N. 3 - LUG/AGO 1988
Sped. in abb. post. Gr. III/70
L. 14.000
68 000
AMIGA
BYTE
by Elettronica 2000
IN REGALO LE LENTI STEREO
TELEMATICA
AMIGO MODEM
STEREOSCOPIA
IL MONDO IN 3D
Tips & Tricks
L'assegnamento delle Variabili
Giochi & Avventure
Novità e Soluzioni
Come Manipolare i File
RENDI
PIÙ POTENTE
LA TUA STAMPANTE
FLIGHT SIMULATOR II
DAL DECOLLO
ALL'ATTERRAGGIO
AMIGA

N. 3 - LUGLIO/AGOSTO 1988

Direttore
SIRA ROCCHI

Direzione Editoriale
MARIO MAGRONE

Direzione tecnica
ENRICO DI ZENOBIO

Segreteria di Redazione
SILVIA MAIER

Grafica
NADIA MARINI

Fotografie
MARIUS LOOK

Disco a cura di
VITTORIO FERRAGUTI

Hanno collaborato: Giulio Bonifazi, Alessandro Bossi, Luca Brigatti, Marco Brovelli, Carlo Cattoni, Leonardo Fei, Vittorio Ferraguti, Alberto Ranzani, Ricky Sword.

Redazione
C.so Vitt. Emanuele 15
20122 Milano
tel. 02/706329

Amministrazione, Redazione, Pubblicità, Arcadia srl: C.so Vittorio Emanuele 15, 20122 Milano. Fotocomposizione: Composit, selezioni colore e fotolito: Eurofotolit. Stampa: Garzanti Editore S.p.A. Cernusco S/N (MI). Distribuzione: SO.DI.P. Angelo Patuzzi spa, Via Zuretti 25, Milano. Amiga Byte è un periodico mensile registrato presso il Tribunale di Milano al n. 215 il 29 marzo 1988. Resp. Sira Rocchi. Spedizione in abbonamento postale Gr. III/70. Pubblicità inferiore al 70%.

IL MENU

E SUL DISCHETTO...

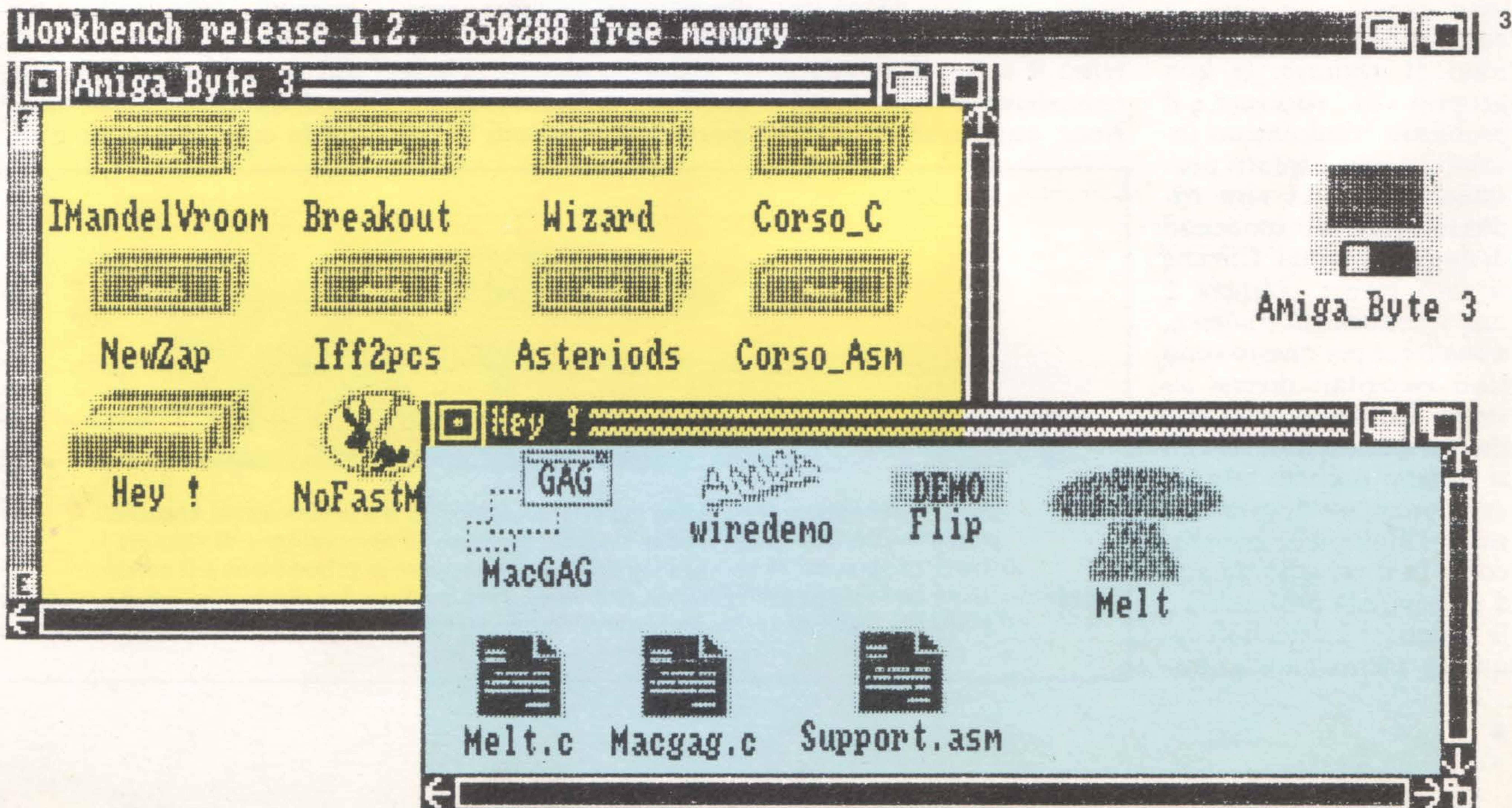
Se vi siete trovati in difficoltà con l'utilizzo dei comandi del sistema operativo di Amiga e avete bisogno di una mano, vi sarà di valido aiuto *WIZARD*, uno dei programmi presenti sul dischetto allegato a questo fascicolo di Amiga Byte. Grazie alla sua semplicità d'uso, rende possibile lavorare in ambiente AmigaDos senza bisogno di digitare comandi o studiare manuali. Con il solo utilizzo del mouse potete attivare le principali funzioni del CLI, visualizzare il contenuto di una directory, copiare files da un disco all'altro, eseguire programmi ed altro ancora.

Se invece il vostro intento è quello di esaminare il codice di un programma per studiarlo o modificarlo, *NEWZAP* fà per voi. Questo potentissimo file editor può essere usato anche, tra le altre cose, per inserire nei programmi i codici esadecimali pubblicati nella rubrica «Tips & Tricks» ed ottenere vite infinite in alcuni tra i più popolari giochi.

Direttamente dalle sale-gioco poi, ecco *ASTERIODS*, un classico videogame spaziale che sfrutta a fondo la velocità e le capacità di Amiga, con effetti sonori digitalizzati e grafica in alta risoluzione. Per mettere alla prova la vostra abilità e quella dei vostri amici potete invece usare *IFF2PCS*, che scompone come un puzzle una qualsiasi schermata grafica in formato IFF. Potete realizzare rompicapo sempre diversi e con difficoltà crescente, usando anche disegni creati da voi. Il terzo gioco di questo mese è *3D BREAKOUT*, un'inconsueta e spettacolare variante tridimensionale del celeberrimo gioco scopo del quale è abbattere un muro di mattoni «armati» di una racchetta e di una pallina.

Le capacità grafiche di Amiga non sono state trascurate: potete esplorare il magico mondo dei grafici di Mandelbrot e dei frattali con *IMANDELVROOM*, un programma dal nome difficile da pronunciare ma facile da usare, che disegna e permette di salvare su disco curve multicolori basandosi su parametri specificati da voi.

Infine, abbiamo raccolto in una directory chiamata *HEY*! alcuni programmi dimostrativi delle capacità spettacolari di Amiga: *MACGAG*, per emulare la gestione delle finestre di un Apple Macintosh; *FLIP*, un programma che vi farà girare la testa, letteralmente; *WIREDEMO*, dimostrazione di animazione grafica tridimensionale; *MELT*, un altro incredibile demo impossibile da descrivere: bisogna vederlo!

Per risparmiarvi poi la fatica della digitazione ed eliminare il rischio di errori di trascrizione, abbiamo incluso in due directory sul disco i listati dei corsi di C ed Assembler presenti in questo fascicolo.

Il mondo in 3D

È un vecchio sogno dell'uomo quello di visualizzare, su di una superficie bidimensionale, il mondo tridimensionale che lo circonda. Forse, con qualche piccolo accorgimento, possiamo dire di esserci riusciti.

di LUCA BRIGATTI

Da quando esiste il disegno, gli uomini si sono posti il problema di come rendere su una superficie bidimensionale il mondo tridimensionale che ci circonda. Le due dimensioni del foglio di carta, della tela o dello schermo di un computer possono riprodurre soltanto due dimensioni delle tre dello spazio (considerando solo lo spazio ed accantonando per il momento la quarta dimensione, il tempo). Dai primi graffiti in cui gli oggetti lontani venivano convenzionalmente rappresentati più in alto di quelli vicini e dopo secoli di tentativi sono state trovate alcune soluzioni, a volte geniali, che permettono di aggirare, se non proprio di risolvere, il problema, tecnicamente insolubile: non è infatti possibile ricreare la *vera* tridimensionalità disponendo di due dimensioni. Ciò che si può invece ottenere è una *sensazione di tridimensionalità*, e per questo sono stati escogitati diversi sistemi. Per capire il principio sul quale questi sistemi si basano occorre fare alcune premesse di carattere neurofisiologico, ovvero: come fa il cervello umano a percepire la profondità e le distanze? I sistemi di cui questo meraviglioso strumento dispone sono fondamentalmente tre; descriveremo ora brevemente i primi due. Non ce ne vogliano i lettori se trascureremo il terzo; esso infatti richiederebbe una descrizione eccessivamente tecnica, senza peraltro avere riscontri pratici su quanto verrà detto in seguito riguardo alla tridimensionalità su *Amiga*.

Il primo sistema si basa fondamentalmente sui rapporti di dimensioni fra gli oggetti e sulla constatazione che, aumentando la distanza dall'osservatore, le dimensioni apparenti di un oggetto diminuiscono. Ad esempio, se abbiamo di fronte un uomo ed al suo fianco un'automobile che arriva come altezza alle ginocchia di costui, possiamo dedurre che l'automobile si trova alcuni metri dietro all'uomo. Se invece si verificasse che l'uomo appare alto come una ruota della vettura, ecco che dedurremmo che è l'uomo a trovarsi alcuni metri distante dalla macchina.

In realtà non occorre che noi si faccia coscientemente delle deduzioni; vedendo una delle scene di cui sopra *sapremmo* immediatamente come stanno le cose, grazie ai con-

Note le dimensioni reali di due oggetti, un confronto fra le dimensioni apparenti permette di capire quale dei due si trovi più vicino all'osservatore e di valutare le distanze reciproche. In questo caso l'automobile appare in primo piano e il corridore distante. In un certo senso perciò questa immagine, certamente a due dimensioni perché stampata sul piano del foglio di carta che toccate, dà una prima sensazione di tridimensionalità.

FOTO FRANCO GENGOTTI

Un'immagine stampata... male ad arte: provate a vederla con gli occhiali (da ritagliare dalla contro-copertina) ed i "vetrini" colorati che vi abbiamo regalato! Risulterà a tre dimensioni!

fronti che il cervello fà istantaneamente.

Un sistema a questo analogo è quello delle sovrapposizioni: se l'uomo fosse stato davanti alla macchina, parte della macchina stessa sarebbe stata nascosta, dal che conseguirebbe immediatamente che l'uomo è più vicino della macchina.

LA PROSPETTIVA

Sul principio enunciato in precedenza, e cioè che le dimensioni apparenti di un oggetto diminuiscono all'aumentare della distanza, si basa la tecnica di disegno nota come *prospettiva*.

Essa può sembrare scontata ma in realtà non lo è, se si pensa che bisogna aspettare il '400 per una definizione rigorosa delle regole geometriche che governano la prospettiva (anche se pare che già notevoli risultati avessero ottenuto i matematici greci, risultati andati poi quasi completamente perduti). Su questa impostazione si basano i dipinti rinascimentali e la stessa fotografia che, riprendendo la realtà così com'è, tiene «naturalmente» conto dei dettami prospettici.

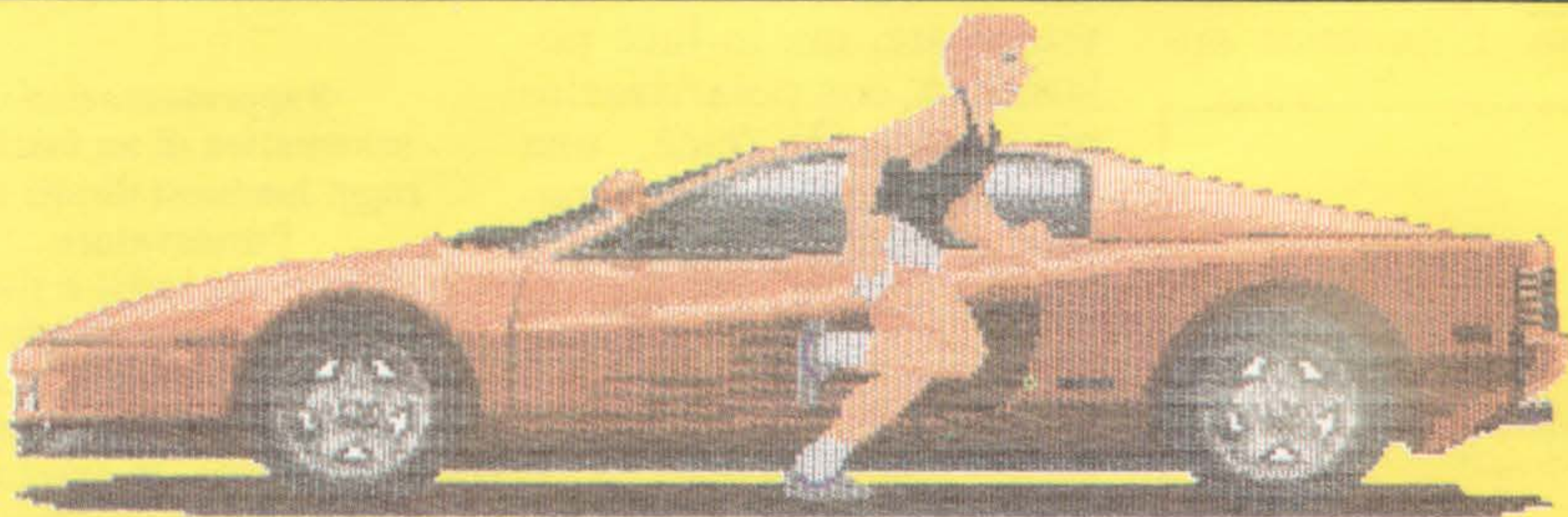
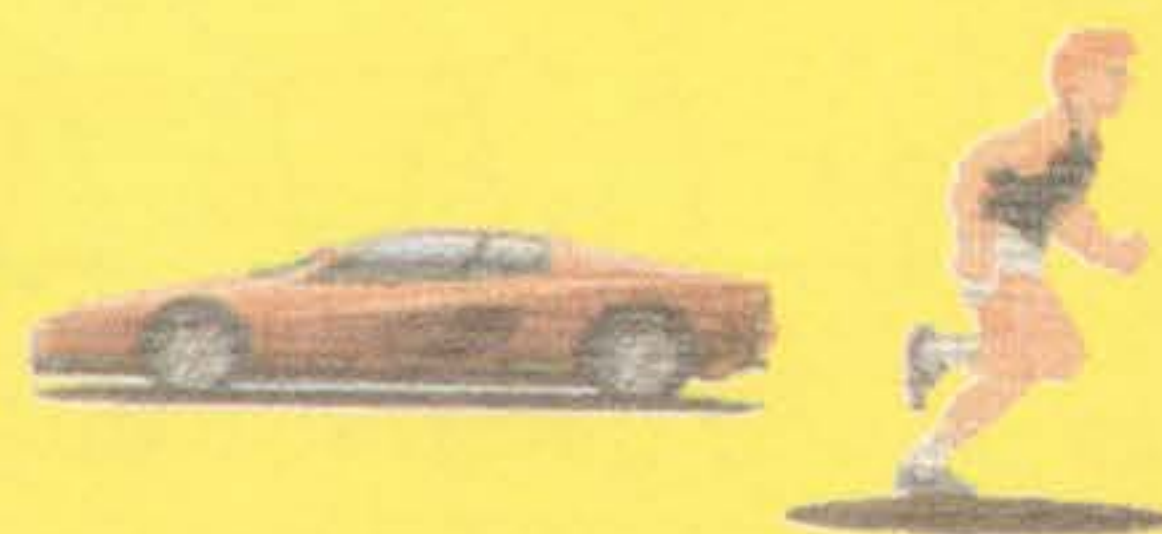
È questo il principio su cui si basano programmi di disegno tridimensionale fra i quali il più noto è forse «Sculpt 3D». Tale programma permette di disegnare uno o più oggetti tramite la loro rappresentazione in proiezioni ortogonali (qui compare ancora la limitazione del foglio a due dimensioni), di specificare la distanza dell'osservatore, il punto di vista, le sorgenti luminose ed altri parametri.

Quindi il programma, tramite sofisticati calcoli matematici che adottano una tecnica detta di *Ray Tracing*, riproduce la scena tridimensionale come apparirebbe nella realtà.

Il secondo metodo di cui dispone il cervello è quello della *visione stereoscopica*.

Come ben sappiamo, l'uomo è fornito di due occhi situati sullo stesso asse orizzontale e simmetricamente rispetto ad un asse verticale. Per quanto minima, la distanza fra i due occhi è tale per cui le immagini che essi raccolgono sono diverse, o meglio sono prese da due punti di vista leggermente diversi. Il cervello prende queste due immagini e le fonde in modo da ottenere una sola immagine, ma con il risultato di rendere anche la *sensazione* della profondità.

LA PROFONDITÀ

È possibile dunque riprodurre anche la sensazione di profondità, andare cioè oltre la rappresentazione già tecnicamente corretta della prospettiva? La risposta è affermativa e il principio su cui si basa è abbastanza semplice: se la visione stereoscopica è data dal fatto che i due occhi vedono la stessa immagine da due punti di vista leggermente diversi, basta dare da vedere ai due occhi *due* immagini prese (avete

Quando due oggetti si trovano così vicini da non consentire una discriminazione delle distanze relative soltanto in base alle dimensioni apparenti, è possibile determinare quale dei due oggetti sia più vicino all'osservatore quando uno dei due nasconda parzialmente l'altro. In questo caso si deduce che il corridore si trova davanti, sia pure di poco, all'automobile.

Le dimensioni apparentemente maggiori del corridore lasciano dedurre che sia quest'ultimo il più vicino all'osservatore.

GLI AMICI DELLA TERZA DIMENSIONE

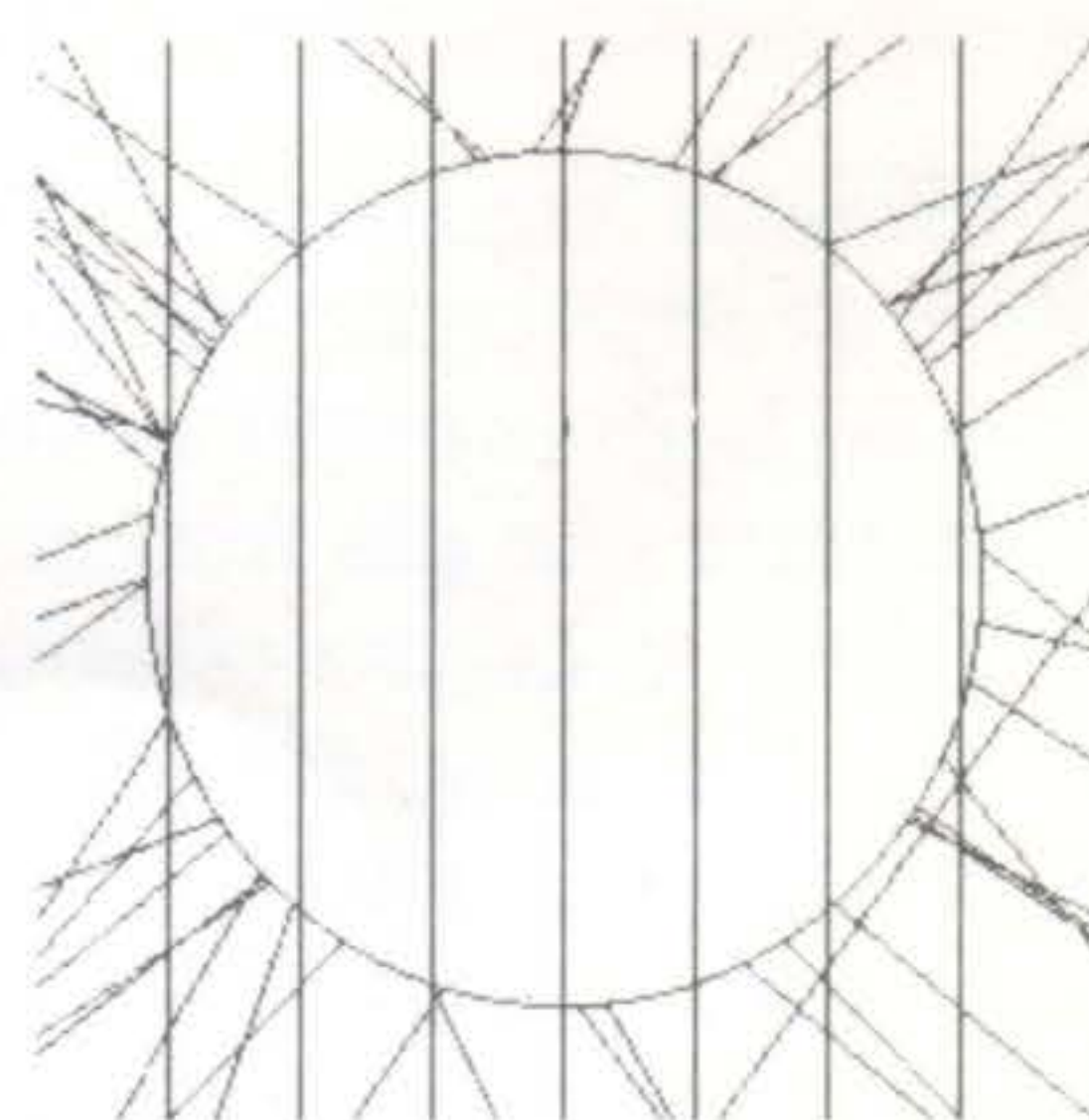
La sigla A.S.I. sta per Associazione Stereoscopica Italiana; presieduta da Franco Gengotti, riunisce in tutta Italia gli appassionati di fotografia, cinema e televisione tridimensionale ed appartiene all'Unione Stereoscopica Internazionale. Scopo dell'associazione è studiare e diffondere il 3D organizzando mostre, conferenze, serate e pubblicando un bollettino periodico di informazione tridimensionale. Fornisce inoltre consulenze per realizzazioni editoriali 3D, ma anche e soprattutto appoggio agli appassionati che vogliono cimentarsi in questa tanto antica ed affascinante tecnica. Per mettersi in contatto con l'A.S.I. è sufficiente telefonare nelle ore serali allo 039/734144, oppure scrivere direttamente ad A.S.I., viale Campania 66, 20052 Monza (MI). L'associazione è comunque a disposizione di chiunque abbia da soddisfare una curiosità... tridimensionale.

già indovinato?) da due punti di vista leggermente differenti.

Il problema però a questo punto si fa pratico; se ci limitassimo a sovrapporre due immagini quasi uguali, *entrambi* i due occhi vedrebbero *entrambe* le due immagini, con il risultato di dire addio alla tridimensionalità e vedere una foto molto confusa. Pertanto occorre che, per usare una parafrasi, non sappia l'occhio destro quello che vede il sinistro; la fusione delle due immagini deve avvenire allorché gli impulsi nervosi arrivano al cervello e non prima. Il sistema migliore è forse quello di usare un apposito visore doppio per diapositive stereo: la diapositiva di destra verrà vista dall'occhio di destra, e la sinistra dal sinistro. I *view masters* si basano su questo principio, se non ne avete mai visti non sapete cosa vi siete persi!

Altro sistema, più elaborato ma più comodo, sfrutta immagini in luce polarizzata. Si sa che ogni singolo raggio di luce può essere considerato un'onda sinusoidale avente una frequenza ed un'ampiezza definite. La frequenza corrisponde visivamente al colore, l'ampiezza all'intensità luminosa. Ad esempio, nell'ambito dello spettro visivo (cioè di quel campo di frequenze corrispondenti alla luce visibile), le frequenze più alte corrispondono alla luce violetta, le più basse al rosso. Scendendo con le frequenze si incontrano i raggi infrarossi e le onde radio; salendo oltre lo spettro visibile si arriva all'ultravioletto, raggi X e raggi gamma. Ma c'è un'altra caratteristica che distingue le singole onde luminose: il piano su cui viaggiano. L'ipotetica onda in figura D si trova su un piano corrispondente al foglio di carta sul quale è disegnata, ma le inclinazioni dei piani nello spazio possono essere infinite.

Rappresentazione schematica di un'onda luminosa: In ascisse il tempo e in ordinate l'intensità. L'intensità rappresenta la luminosità del raggio di luce e la frequenza (numero di cicli in un secondo) il colore. Il piano del raggio coincide con il foglio su cui è disegnato.

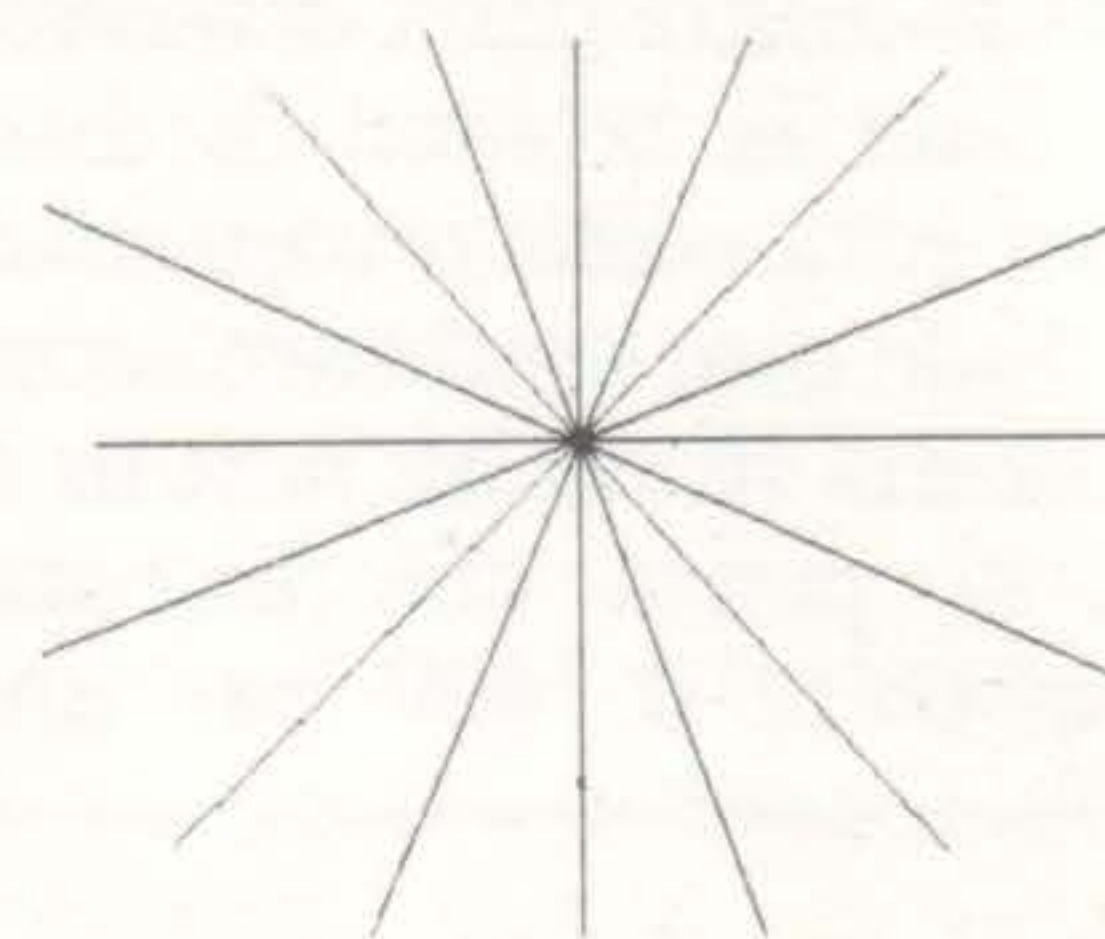
Un filtro polarizzatore ha la caratteristica di rendere *polarizzato* un fascio di luce; ciò significa che tutte le onde luminose, dopo aver attraversato il filtro, viaggiano su piani paralleli, o meglio, il filtro lascia passare solo le onde che viaggiano su piani paralleli. Ovviamente, semplicemente inclinando il filtro si possono selezionare fasci di luce di angolo desiderato.

Rappresentazione schematica di un filtro polarizzatore verticale attraverso cui passa un fascio di raggi con asse diversi e centro non in comune. Si noti che il filtro polarizzatore lascia passare soltanto i raggi verticali.

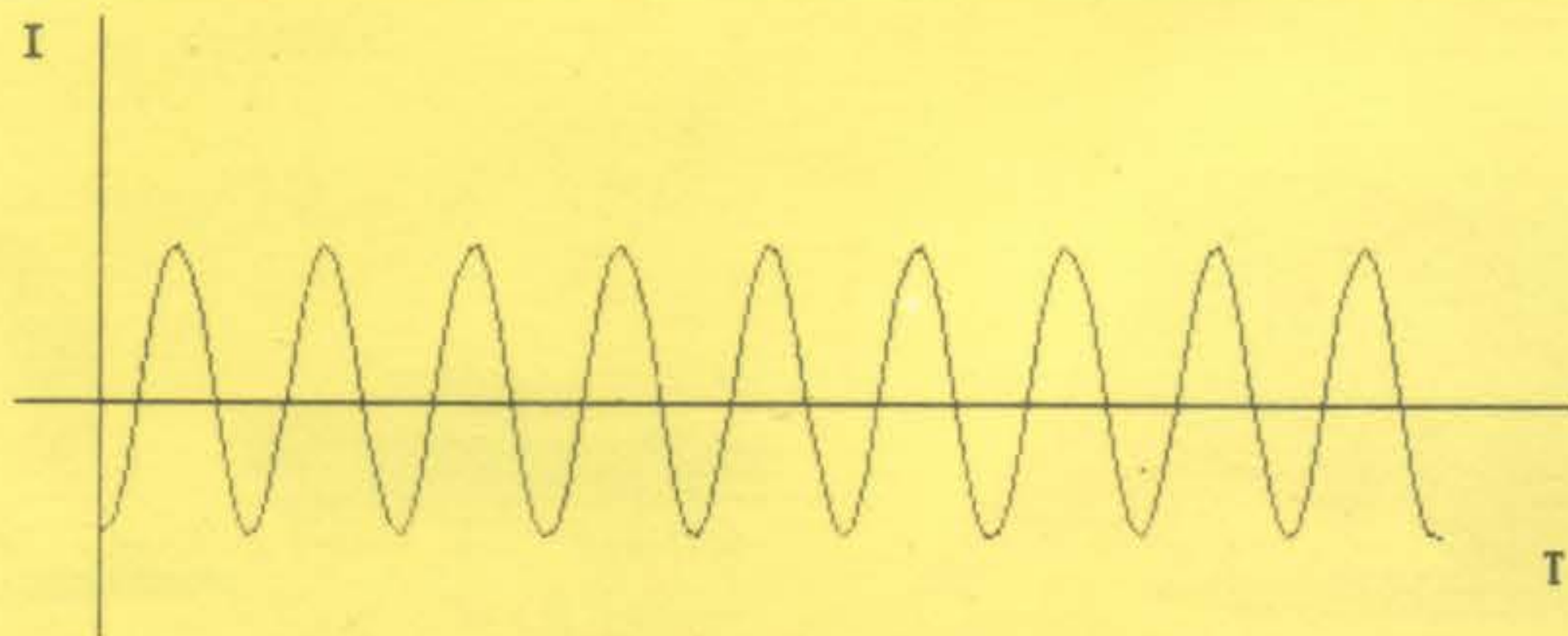
È facile capire come questa caratteristica possa essere sfruttata per la stereoscopia: in pratica il sistema consiste nel proiettare su uno schermo apposito le due immagini sovrapposte, ma in luce polarizzata, con polarizzazioni ortogonali (cioè, una con polarizzazione orizzontale e l'altra verticale) e fornire allo spettatore un paio di occhiali anch'essi polarizzati, in modo tale che solo l'immagine di destra possa raggiungere l'occhio destro (perché immagine e filtro hanno la stessa polarizzazione) e l'immagine di sinistra il sinistro.

Su questo principio si basano i film stereoscopici di moda negli anni '50 e dei quali un recente esempio è «Lo Squalo 3». Una variante di questo sistema, utilizzabile anche per le stampe e per le immagini di *Amiga*, ma dai risultati leggermente peggiori, è quella degli occhialini colorati; in questo caso la selezione delle due immagini avviene grazie alle lenti di colori complementari, cioè tali per cui un'immagine passata attraverso un filtro rosso può essere raccolta solo dall'occhio con davanti il filtro dello stesso colore, mentre viene fermata dall'altro filtro.

Le immagini presentate in questo articolo sfruttano questo principio; si noti come queste, viste ad occhio nudo, appaiano estremamente confuse. Esse assumono tutta la loro bellezza stereoscopica non appena vengono indossati gli occhialini fatidici.

BREAKOUT 3D

Lo stesso gioco presentato nel disco allegato ad *Amiga Byte*, «Breakout 3D», si giova dell'effetto stereoscopico mediato dai filtri colorati per rendere la sensazione di giocare a

Rappresentazione schematica di un fascio di raggi luminosi diretti verso l'osservatore. Il piano dei raggi è visto di profilo, pertanto appare come un segmento di lunghezza pari all'intensità di quell'onda. In questo caso sono stati rappresentati raggi con piani aventi un centro comune.

Breakout all'interno di una stanza. Anche qui il gioco

risulta impossibile se non si possiedono gli occhiali colorati. Essi purtroppo non sono facilmente reperibili; il modo migliore per ottenerli è probabilmente quello di ordinarli direttamente a Franco Gengotti, presidente dell'*Associazione Stereoscopica Italiana* (Vedi riquadro) nonché accanito fan di *Amiga*: inviando L. 3.000 in francobolli vi verranno spediti gli occhiali a stretto giro di posta.

Se si vuole (è un invito!) si potranno sperimentalmente utilizzare i due filtri colorati che vi abbiamo regalato in questo stesso numero. Basterà, con un po' di pazienza, costruirsi una montatura di carta spessa seguendo la traccia indicata nella controcopertina, sagomando e incollando i due filtri colorati. Con tali occhiali sarà interessante verificare l'effetto profondità nelle foto «strane» di queste stesse pagine.

Il piccolo sforzo che occorre fare per procurarsi gli occhiali però vale senz'ombra di dubbio lo spettacolare effetto che si ottiene dalla stereoscopia.

Come fare dunque per creare delle immagini, non solo tridimensionali, ma anche stereoscopiche? Con il buon vecchio sistema della macchina fotografica; basta scattare due diapositive con la fotocamera spostata di due o tre centimetri (di più, se le immagini da riprendere sono molto lontane) fra una diapositiva e l'altra. Se si desidera invece ottenere fotografie tridimensionali, occorre munirsi di occhialini colorati ed effettuare il primo scatto con il filtro verde scuro e poi, *senza far avanzare la pellicola*, fare la seconda foto (dopo aver spostato la macchina a sinistra) attraverso il filtro rosso. Il risultato su carta sarà visibile con gli stessi occhialini. Unica avvertenza: fate in modo che l'inquadratura delle due foto sia, per quanto possibile, uguale e, ovviamente, non fotografate soggetti in movimento! E per ottenere immagini di cotal fatta su *Amiga*? Il sistema, grazie a potenti pacchetti di software grafico (Vedi riquadro) è più semplice di quello che si possa pensare. Questi pacchetti consentono di creare figure dotate di prospettiva selezionando tutta una serie di parametri fra i quali il punto di vista. Per ottenere due immagini, destra e sinistra, occorre semplicemente, a parità di tutti gli altri parametri, far eseguire i calcoli specificando per un'immagine un punto di vista e per l'altra un diverso punto di vista situato sullo stesso piano orizzontale ed alla stessa profondità del precedente, ma spostato di qualche centimetro (o anche di più, se gli oggetti risultano essere molto lontani dall'osservatore) a destra o a sinistra, in base ad un principio fondamentalmente identico a quello visto per le diapositive e adottato dai nostri stessi occhi.

Una montagna... stereo: usare gli occhiali speciali.

Terminate le elaborazioni, si otterranno due immagini molto simili; a questo punto occorre scattare una diapositiva del video per ogni immagine ed il gioco è fatto: con un visore per diapositive stereo dal costo modesto (reperibile anch'esso presso Franco Gengotti o da un giocattolaio ben fornito) si potrà esaminare la nostra opera d'arte (perché, checché se ne dica, di arte si tratta) in tutta la profondità della stereoscopia.

FOTO AL VIDEO

Circa la tecnica per fotografare il video, dovete andare un po' a tentativi; usate magari un cavalletto e tempi lunghi (superiori al secondo); non includete nell'inquadratura i bordi: tanto più la ripresa è centrale tanto meno si noterà la distorsione del monitor; sappiate infine che le immagini interlacciate vengono bene quanto le altre, in fotografia. Non scoraggiatevi se inizialmente incorrerete in risultati deludenti; per ottenerne di interessanti occorre infatti un briciolo di esperienza e soprattutto è necessario avere bene in mente i principi che regolano la visione stereoscopica. Comunque, credeteci, la fotografia o il disegno stereoscopico sono molto più entusiasmanti dei loro parenti poveri in 2D.

Cimentatevi nell'impresa ed inviateci i vostri migliori lavori (su dischetto o su diapositiva, accompagnati da una breve documentazione riguardante le tecniche di realizzazione); non è detto che, se sono interessanti, non verranno in futuro pubblicati. ■

Software Express

a cura di GIULIO BONIFAZI

Brevi recensioni di giochi ed utility freschi di giornata. Visti, giocati e valutati per voi.

SUFFICIENTE — DISCRETO — BUONO

OTTIMO!

WIZBALL

OCEAN

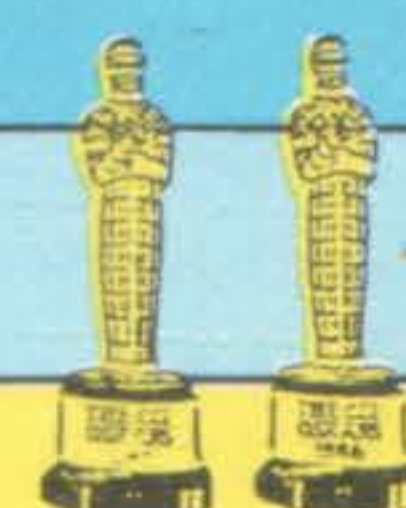
Ottima implementazione del noto gioco pubblicato lo scorso anno dalla inglese Ocean, appena provato su Amiga. La grafica è eccellente, lo scroll del paesaggio perfetto. Tutto si svolge su di un pianeta sconosciuto, infestato da ogni tipo di alieni. Controllerete una simpatica pallina magica che rimbalzerà in continuazione, anche quando le ordinerete di non muoversi. La schermata di apertura è identica a quella vista a suo tempo sulla pubblicità delle riviste specializzate, e non ha nulla a che vedere con quelle di altri computer. La pallina possiede molta inerzia, per cui è necessario stare bene attenti nel controllarla: abbiate sempre presente da che parte è situata la boccuccia, in maniera da decidere al momento giusto la direzione verso la quale andare. Quando la sfera è in aria non è possibile guidarla a destra o a sinistra, ma soltanto ruotarla: appena toccata terra, essa schizzerà via molto velocemente. Tutti gli oggetti immobili del pianeta, quali stelle e puntoni, sono dei potenti materassi che vi respingeranno ogni qualvolta li toccherete. Non vi succederà nulla durante queste collisioni, dal momento che lo spirito del gioco è proprio basato su urti e relativi rimbalzi, ma se non starete sufficientemente in guardia vi troverete in grossi guai. Tutto quello che si muove è infatti dannoso per la vostra salute, pertanto dovrete cercare di colpire qualunque cosa vediate, il che non è facile visto che, anche per i colpi, vale lo stesso discorso fatto per gli spostamenti: ovvero la raffica partirà nella direzione verso la quale la bocca

della pallina sarà rivolta. Ci sono alieni dotati di poteri magici che, una volta disintegrati, svilupperanno delle bolle verdi che dovrete assolutamente prendere (andando a sbattervi contro) perché vi concederanno nuove possibilità di spostamento e di fuoco. In corrispondenza di ciascuna bolla vedrete, in alto, lampeggiare l'opzione selezionabile: se desiderate avvalervene tenete premuto il fuoco e vi accorgerete subito dei vantaggi. Infine, provate ad entrare nelle numerose buche del pianeta; vi troverete su di un altro livello e, se tenterete di nuovo, scenderete ancora scoprendo nuovi quadri e nuove difficoltà. Va da sé che conviene disegnarsi una mappa.

SOCCER SUPREMO

S.P. ASSOCIATES

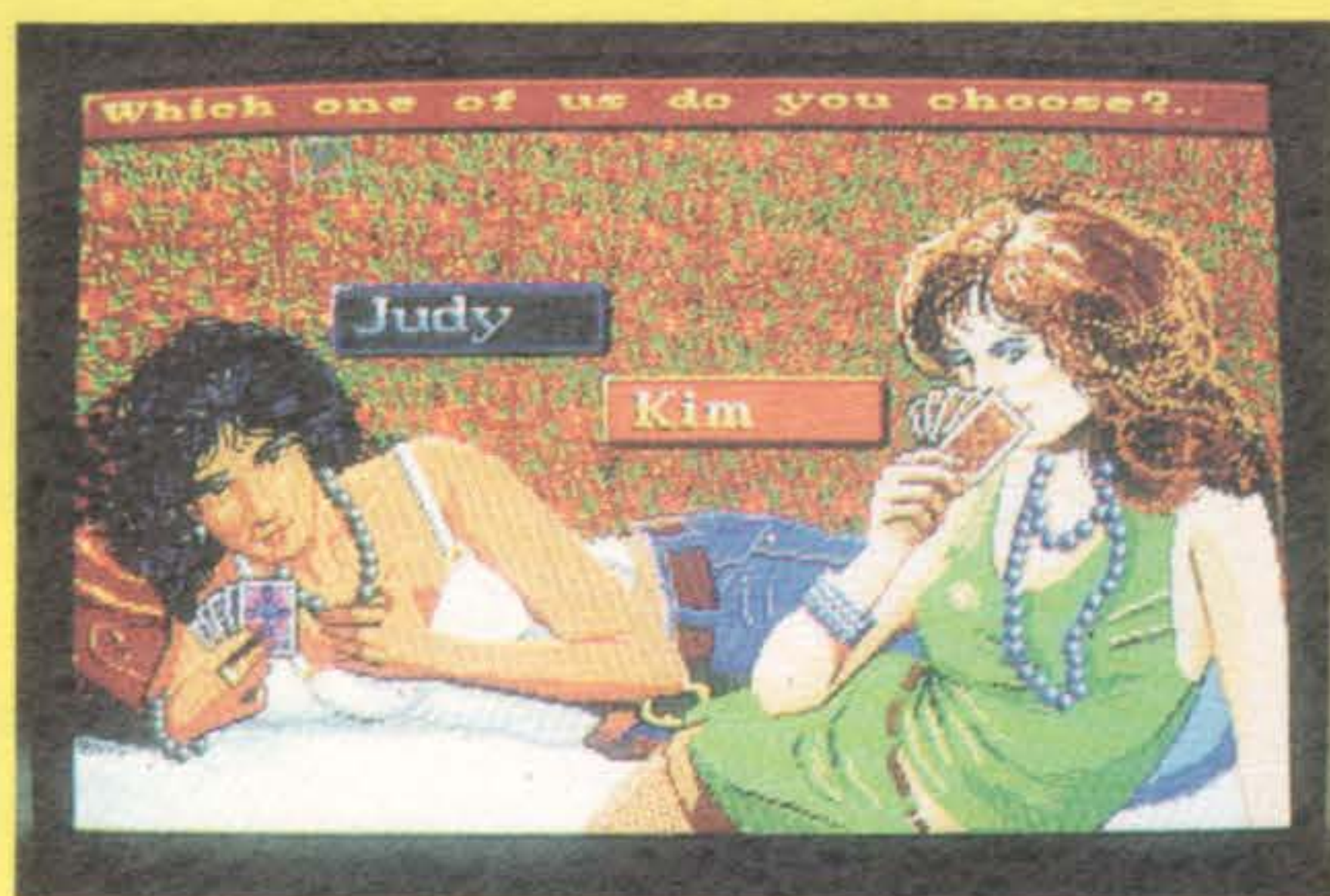
I patiti del calcio «ragionato», quelli che non si immedesimano tanto nei giocatori quanto piuttosto nei direttori tecnici, troveranno in questo software pane per i propri denti: dovranno infatti formare una squadra, e progredire nel campionato inglese.
Il gioco non è quindi un'arcade per smanettoni, ma una vera simulazioe, per così dire «gestionale»: l'andamento del vostro team dipenderà infatti solo da come imposterete la squadra nelle varie partite, da come gestirete i vostri giocatori, la vostra panchina ed i vostri soldi.
All'inizio il compito sarà quello di formare una squadra decente con il portafoglio che avete a disposizione, scegliendo i giocatori a seconda del ruolo, dell'età, del prezzo e dell'abilità. In seguito, se otterrete buoni risultati, cresceranno le loro quotazioni (e le vostre!) e potrete effettuare movimenti di mercato.
Lo schermo di gioco è senza fronzoli, quasi un Cli, e l'input dei comandi è da tastiera: il programma non accetta input sbagliati e non vi permette di competere in campionato se la vostra squadra non è completa.
Approntata la formazione per l'incontro, potrete affrontare il campionato; mentre attenderete l'esito verrete allietati da un'immagine calcistica digitalizzata (ce ne sono diverse).
L'esito dell'incontro sarà stabilito dalle forze in gioco, dal «fattore campo», e dal computo delle probabilità stabilito dal computer, che ve lo mostrerà prima della partita, sì che possiate orientarvi meglio nelle scelte. ■

FROST BYTE

MIKROGEN

Ennesima conversione inglese di un vecchio successo, protagonista delle hit parade di qualche anno fà, ora disponibile anche su Amiga. Nulla da dire sulla implementazione del gioco, quasi una copia carbone dell'originale; ci mancherebbe che non si riuscisse a portare sul 68000 un giochino scritto per lo Z80 o per il 6502! Naturalmente, a chi non è piaciuta la versione originale non potrà piacere neanche questa ma, tutto sommato, il gioco è divertente.
Guiderete un tubetto cilindrico che per muoversi è costretto a rotolare su se stesso e che è anche in grado di saltare; se vi impossesserete delle munizioni, potrà financo sparare. Sarà bene provvediate subito a rifornirvi di colpi, altrimenti non potrete neppure vedere il secondo quadro. Se non vi muoverete appena inizia il gioco, troverete le munizioni proprio a fianco a voi: andatevi contro e le avrete immagazzinate. Ora potete scendere, far fuori la strega ed incamminarvi verso l'ignoto. Il secondo quadro, adiacente al primo, è già molto più difficoltoso da attraversare, perché alcune simpatiche gocce cadono regolarmente impedendovi di proseguire. Sarà necessario, qui, cronometrare al millesimo di secondo i vari passaggi: la pazienza però ha un limite e vi capiterà sovente di strapazzare il joystick nel vano tentativo di superare un determinato punto. Non preoccupatevi più di tanto: in fondo è sufficiente fare un minimo di pratica per memorizzare tutti i passaggi più difficili, e da quel punto in poi vi divertirete ad andare in giro con il vostro cilindretto sparando all'impazzata.

STRIP POKER

PLAY HOUSE

Tanto per non cambiare, cercate di spogliare una ragazza vincendo i suoi abiti in una partita a poker: ogni pezzo del suo vestiario viene pagato cento dollari.
Le ragazze sono due, una rossa di capelli, l'altra con la chioma... blu. Scelta, all'inizio, la nostra sfidante, ce la troveremo di fronte in posa provocante e pronta alla sfida maliziosa.
Cominciamo con le note positive: l'azione di gioco è discretamente veloce ed immediata ed il computer non è proprio un mago, il che permette di vincere abbastanza facilmente (un pregio all'inizio, che però non giova alla «longevità» del gioco perché, una volta spogliate entrambe le fanciulle, l'interesse cala con la curiosità soddisfatta). Parere negativo per la grafica, poco accurata, e pollice verso sulla completa assenza di suoni.
Data l'estrema facilità con la quale Amiga può digitalizzare le immagini, ci si chiede perché si insista ancora nel proporre programmi di questo genere risolti con semplici disegni: se lo scopo di giochi come questo è di lustrare gli occhi degli affezionati del genere sulle beltà muliebri rese da una manciata di pixel sullo schermo, più logico sarebbe che questi fossero tratti da una persona in carne ed ossa! Aspettiamo con ansia (slurp!) uno strip poker degno di questo nome e della grafica digitalizzata del nostro computer. Questo della Play House è dunque un'ulteriore occasione mancata. E, in ogni caso, ne esistono in commercio di decisamente migliori. ■

CUBE MASTER

ASDG

Il nome del programma rende già l'idea della forma che avranno gli oggetti protagonisti del gioco, ovviamente dei cubi. Apparentemente l'impatto iniziale non è dei migliori, nonostante la schermata di presentazione sia interallacciata e disegnata piuttosto bene. Il quadro di gioco è infatti molto semplice e ridotto al minimo. Gli unici indicatori presenti riguardano le solite cose: il tempo rimasto, il punteggio e la barra dell'energia; il resto è occupato da un'area rettangolare nera, l'arena in cui il nostro personaggio, un astronauta munito di uno zaino spaziale (ricordate Jet Pac?), deve distruggere dei cubi colorati nel minor tempo possibile. Il meccanismo del gioco è simile a quello del famoso Asteroids: il cubo, inizialmente grosso, se colpito si divide in altri cubi più piccoli. Il controllo dell'astronauta è piuttosto difficoltoso; il meccanismo di spinta essendo vettoriale, colpire i cubi più piccoli non è semplice. Per fortuna si può scegliere, all'inizio del gioco, l'opzione «Impara a volare», grazie alla quale è possibile fare un po' di pratica con il joystick (ma il protagonista può essere mosso anche con i tasti cursore). La grafica è scarna ed essenziale, ed il gioco in sé non è particolarmente originale; ma la velocità dei movimenti e la difficoltà nel padroneggiarli lo renderanno gradito ai patiti dello smanettare. Certo è che, appena vi sfuggirà il controllo dell'astronauta, tentare di riprenderlo sarà davvero arduo, perché il nostro eroe inizierà ad andare alla velocità della luce ed a sbattere come una pallina impazzita contro tutti i bordi dell'arena.

FLIGHT PATH 737

La Anco (ex Anirog) prosegue la sua politica di conversione per Amiga di giochi realizzati in precedenza per il C64. Questa volta tocca a Flight Path 737, programma che il manuale indica come simulazione di volo. Lo schermo di presentazione ci mostra un jet in atterraggio in una suggestiva atmosfera quasi romantica, con una buona musica digitalizzata: e questo è, purtroppo, probabilmente il momento migliore di tutto il gioco. Possiamo poi scegliere tra più livelli di difficoltà, che vanno da «Solo Flight» ad «International Licence»; operata la scelta, il video ci mostra una serie di quadranti ed una finestra piuttosto angusta che dovrebbe rappresentare il parabrezza, dalla quale si vede la pista di decollo; il realismo del cruscotto è piuttosto modesto. Tutta la simulazione consiste nel decollare, prendere rapidamente quota in modo da sorvolare delle montagne, ed atterrare su di una pista al di là delle stesse; le crescenti difficoltà comportano monti più alti, piste più corte e la possibilità di subire venti trasversali o incendi ai motori, che vanno spenti il più rapidamente possibile. Tutto qui. I comandi vanno impartiti con il mouse: questo si trasforma, secondo quella che è ormai una consuetudine, da puntatore in volantino mediante la pressione del tasto destro. I parametri sono: decollo a 190 nodi, rientro dei flaps entro i 200 nodi, sollevamento del carrello sopra i 300 piedi, salita veloce, orientamento con la pista di atterraggio, discesa, giù i flaps ed il carrello (con le stesse limitazioni di velocità e quota del decollo), atterraggio. Il tutto in meno di 5 minuti e con una grafica, seppure di chiara derivazione digitalizzata, per niente realistica e molto lenta. Se il Flight Path 737 deve proprio essere definito un simulatore, va detto almeno che è adatto a chi è alle primissime armi; non si tratta di una simulazione semplicissima, ma è decisamente troppo limitata.

ANCO

BMX SIMULATOR

CODE MASTERS

Uno sport, quello delle Bmx, molto popolare nei paesi anglosassoni, che consiste nel correre su campi da cross con biciclette appositamente costruite per questo genere di gare. Dedicato principalmente ai giovanissimi, riscuote consensi anche tra i ragazzi più grandi. Questa proposta è una conversione economica dedicata all'Amiga del gioco Bmx Simulator, uscito tempo fà nella riuscita versione per il C64 e che ebbe un discreto successo. I fondali hanno mantenuto lo stesso stile, così come gli sprite dei giocatori: entrambi però sono stati notevolmente migliorati. Per sottolineare la parentela con la precedente versione, anche la musica d'accompagnamento dei titoli ha lo stesso motivo.
La visione delle varie piste del campo di gara è dall'alto: si può gareggiare contro il computer o contro un amico, scegliendo nello schermo di presentazione i controlli.
La lunghezza della gara può essere fissata fino ad un massimo di sette percorsi diversi, di difficoltà crescente; non è necessario vincere sempre, ma occorre qualificarsi almeno per la gara successiva, completando i giri entro il tempo massimo.
L'azione di gioco, soprattutto nella competizione a due, è decisamente accattivante, ed occorre molta pratica per poter padroneggiare i movimenti del proprio pedalatore. Notevoli le possibilità offerte dall'«action replay» e dallo «slow motion», permessi alla fine di ogni percorso, utili per rivedere una gara esaltante o per rendersi conto dell'errore commesso.
Nelle collisioni tra lo sprite del computer e quello del giocatore, a rimetterci è sempre quest'ultimo: una difficoltà in più che rende il gioco ancor più competitivo. Se volete assistere alla gara tra due giocatori senza muovere un dito, appena terminato il caricamento del programma non toccate niente: pochi istanti dopo avrà inizio il demo.

IN EIGHTY DAYS AROUND THE WORLD

RAINBOW ARTS

Creato da programmatori tedeschi, questo gioco dovrebbe teoricamente essere basato sulle gesta del famoso Mr. Fogg, personaggio ideato da Verne e protagonista del famoso libro «Il giro del mondo in 80 giorni».
Il libro racconta come Mr. Fogg, dopo avere scommesso 20 mila sterline con i soci del suo club, sia riuscito a compiere il giro del mondo affrontando mille peripezie; il programma, che occupa due dischi, dovrebbe renderci protagonisti (simulati) di questa temeraria ed affascinante impresa.
In realtà non si può certo dire che gli sforzi dei programmatori siano stati efficaci, il gioco mancando completamente di qualsiasi velleità.
Si comincia con uno schermo sul quale, mediante un puntatore comandato dal joystick, decideremo se incrementare i nostri averi giocando a carte ad Alto e Basso (perdendo però tempo prezioso), oppure seguiremo su di una mappa il percorso compiuto dal nostro eroe.
Giunti in India, il gioco diventa un platform game nel quale è necessario concludere due fasi nel tempo utile concesso: bisogna attraversare un villaggio indigeno evitando di scontrarsi con le aquile che volano basso e con le bocche di leone; le aquile possono essere uccise con lance sibilanti. Si passa poi in un tempio pieno di pipistrelli, topi e serpenti (originale, eh?); terminati i quadri si torna sulla mappa, il percorso riprende e, per quanto il gioco sia stato provato più volte, non ci risultano altre fasi «arcade»; al terzo tentativo l'avventura è stata felicemente conclusa! I suoni, la grafica e l'interattività del gioco non sono assolutamente all'altezza dell'Amiga: la mancanza di difficoltà e di trama rande poco appetibile un software che, dato il titolo e la mole, avrebbe potuto essere risolto in maniera migliore.

CLIMATE 1.201

PROGRESSIVE PERIPHERALS & SOFTWARE

CLImate è un'utility quasi indispensabile per chi programma in ambiente AmigaDos: essa consente di gestire tutti i comandi più importanti senza doverli digitare per esteso. Grazie a CLImate è possibile eseguirli senza dover premere un tasto, con il solo ausilio del mouse. Lo schermo è diviso in due finestre, denominate «Source» e «Destination», nelle quali appare la lista dei file contenuti in una directory o in un device (Df0:, Df1:, Ram:, Dh0: ...).
Basta selezionare il nome di un file con il mouse e poi clickare sulla funzione desiderata per eseguirla: ad esempio, per copiare un programma dal disco Df0: alla Ram: sarà sufficiente assegnare al primo device la finestra «Source» ed al secondo quella «Destination». Basterà poi clickare sul nome dei files da copiare e sul riquadro «Copy» per portare a termine l'operazione. I principali comandi Dos ai quali CLImate si sostituisce sono: Cd, Dir, Copy, Type, Format, Rename, Delete e Makedir, ed è anche possibile stampare testi e visualizzare file IFF.
La principale innovazione di questa nuova release consiste nella possibilità di disattivare la funzione «.fastdir». Precedentemente, la gestione delle directory Amiga era resa più veloce caricando in memoria la lista dei file e salvandola su disco in un file chiamato «.fastdir», in modo da richiamarla all'occorrenza. Questo file occupa alcuni byte sul disco, e può essere preferibile eliminarlo per risparmiare spazio: un'opzione assente in precedenza, ora implementata in questa versione del programma.

VIDEOSCAPE 2.0

AEGIS

Ecco finalmente l'ultima versione del noto Aegis Videoscape 3D, probabilmente il più completo (ed uno dei più complessi) programma di animazioni tridimensionali per Amiga. La versione 1.0 aveva la risoluzione video tipica dell'NTSC; la 1.1 superava quei limiti con una risoluzione PAL completa di Overscan, ma i colori potevano essere solo 16 o 32. La 2.0 risponde a tutte le critiche riguardanti una limitata scelta dei colori permettendo di lavorare in Overscan interallacciato in modo Hold And Modify!
La gestione dei colori è infine molto sofisticata, ed è possibile un Anti-Aliasing decisamente efficace.
Va detto però che il solo tentare di creare un file ANIM con oggetti anche semplici in modo HAM, conduce ad una inevitabile Guru Meditation se non si hanno ALMENO 4 mega di Ram...
Se le disponibilità finanziarie sono limitate, è anche possibile utilizzare il modo Half-Brite (64 colori in contemporanea); anche in questo caso però la richiesta di memoria aumenta: è chiaro che lo standard minimo da adottare, se si vuole utilizzare il Videoscape in maniera decente, è di almeno 2 megabytes.
Altre chance offerte da questo super programma sono la possibilità di dare le coordinate di quattro fonti luminose, contro la singola della versione precedente; la facoltà di scegliere nella palette i codici dei colori, per poter assegnare la trasparenza anche a colori diversi dal nero.

SIMPLE WRITER

Un word processor, il cui nome dovrebbe essere indicativo delle qualità offerte: un programma abbastanza potente, ma di facile uso. Peccato che, invece, il Simple Writer non sia semplice.
Il programma prevede comunque le opzioni classiche di un word processor ed offre, in più, l'interessante possibilità di ridefinire i tasti funzione con delle macroistruzioni. Risulta poi particolarmente fornito di possibilità di formattazione e di movimento del cursore; in realtà, essendovi pochi shortcuts da tastiera, non è granché comodo dover abbandonare quest'ultima ogni qualvolta si debba utilizzare un'opzione. La mole dei menù potrebbe a prima vista impressionare, ma va detto onestamente che molte delle opportunità offerte sono ridondanti o, peggio, inutili. Da segnalare il fatto che si può tenere in memoria solo un massimo di 5 file.
Inoltre, la potenza di un wp dovrebbe essere misurata valutando la sua capacità di importare la grafica da programmi di disegno; o di caricare o salvare files ASCII o compatibili con altri wp; oppure misurando la sua velocità di caricamento di un file o quella di formattazione di un testo; tutte situazioni nelle quali il Simple Writer mostra qualche pecca.

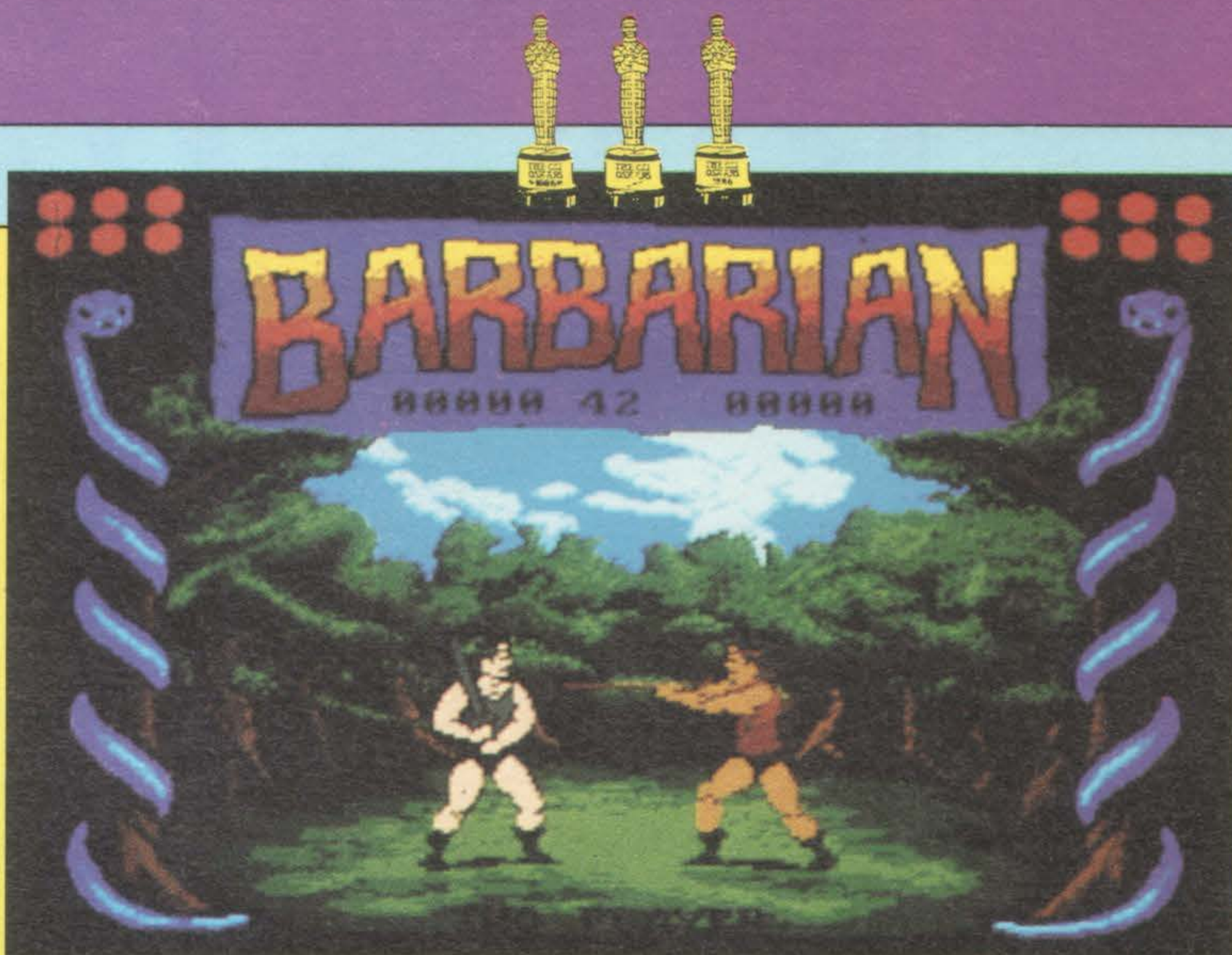
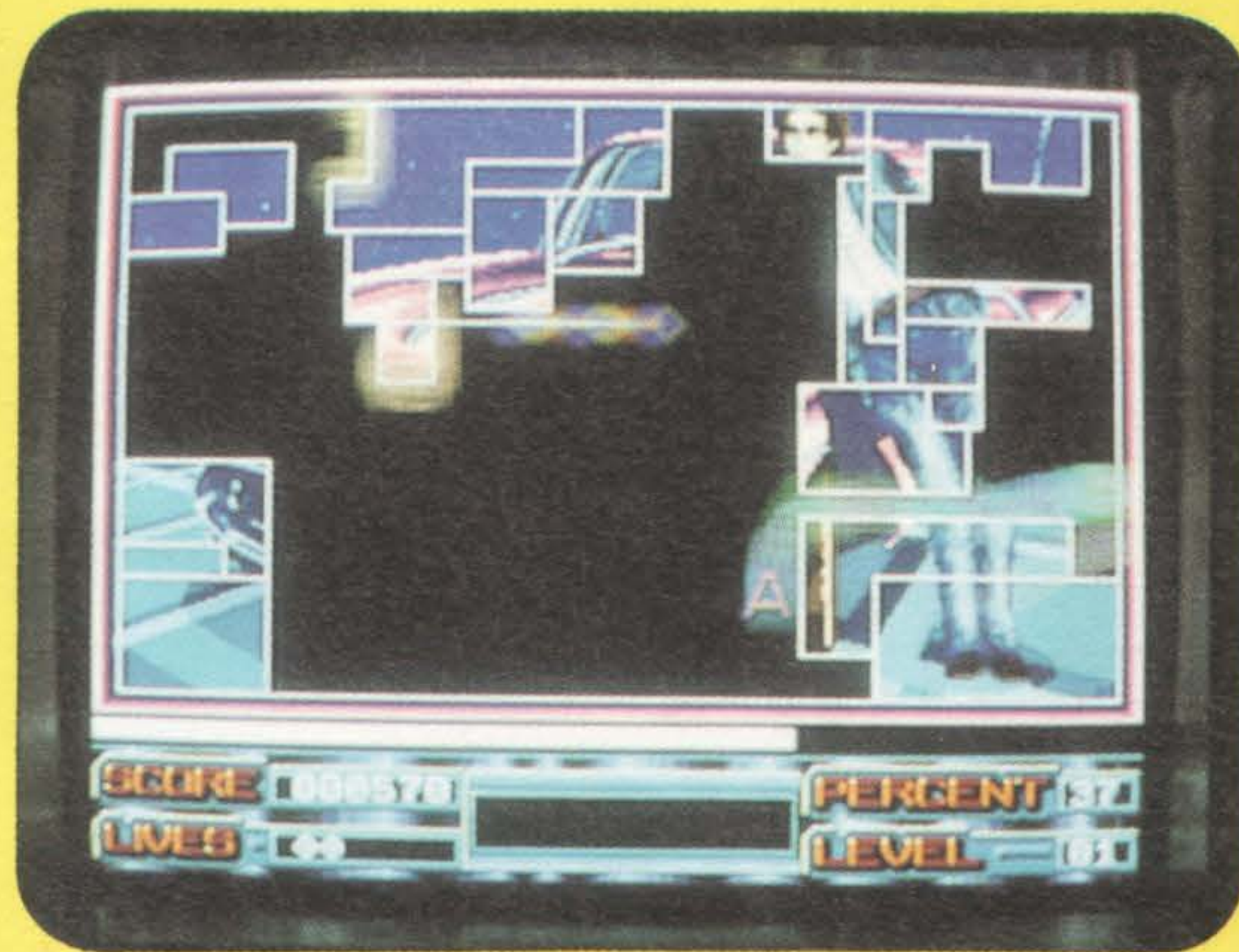
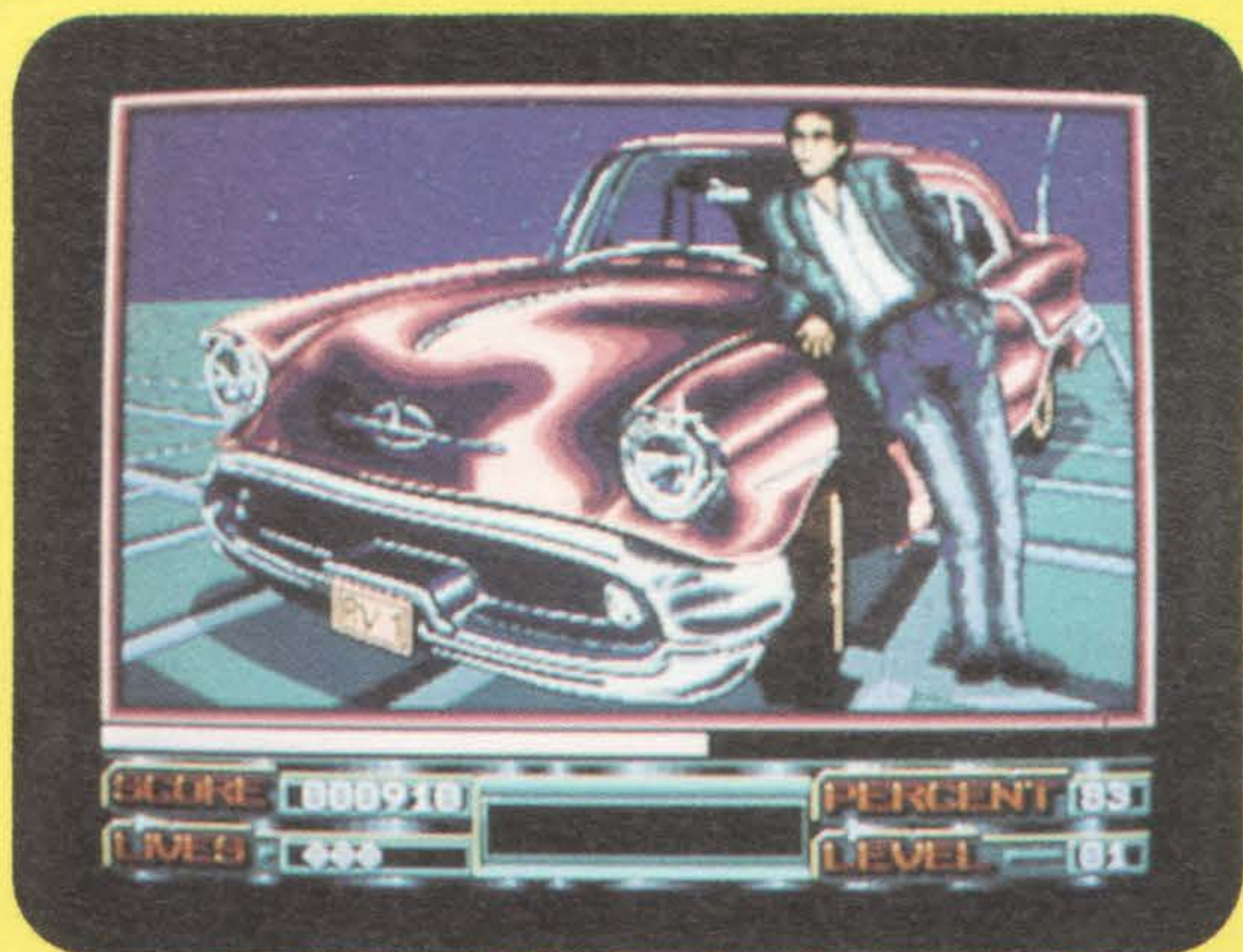
AARGH!

ARCADIA

Terminato il caricamento vi sarà chiesto quale personaggio comandare, se Lizard od Ogre. Il primo è un drago che deve aver mangiato parecchio pepato visto il fuoco che sputa in continuazione, mentre il secondo, per quanto ami il caldo anche lui, somiglia un po' al nostro Hulk e riesce a difendersi anche con i classici pugni. Ci sono numerosi quadri da giocare, dei quali quello con le capanne è soltanto il primo: per accedere ai successivi è necessario distruggere TUTTE le costruzioni visibili sullo schermo, incendiandole. Dirigete lo sguardo sul bersaglio e premete fuoco; il resto è, per così dire, automatico perché, una volta che le fiamme avranno iniziato ad ardere, in capo a pochi secondi la capanna crollerà al suolo, completamente distrutta. L'ultima cosa da portare a termine in questa fase del gioco è impossessarsi di una sfera che appare proprio quando tutto è distrutto. Dopo di che arriva il secondo quadro. Qui le cose sono già un po' più difficili perché, provando a sputare fuoco a destra ed a manca, non succede proprio nulla: tutto sembra essere costruito con materiale ignifugo!

BARBARIAN

PALACE SOFTWARE

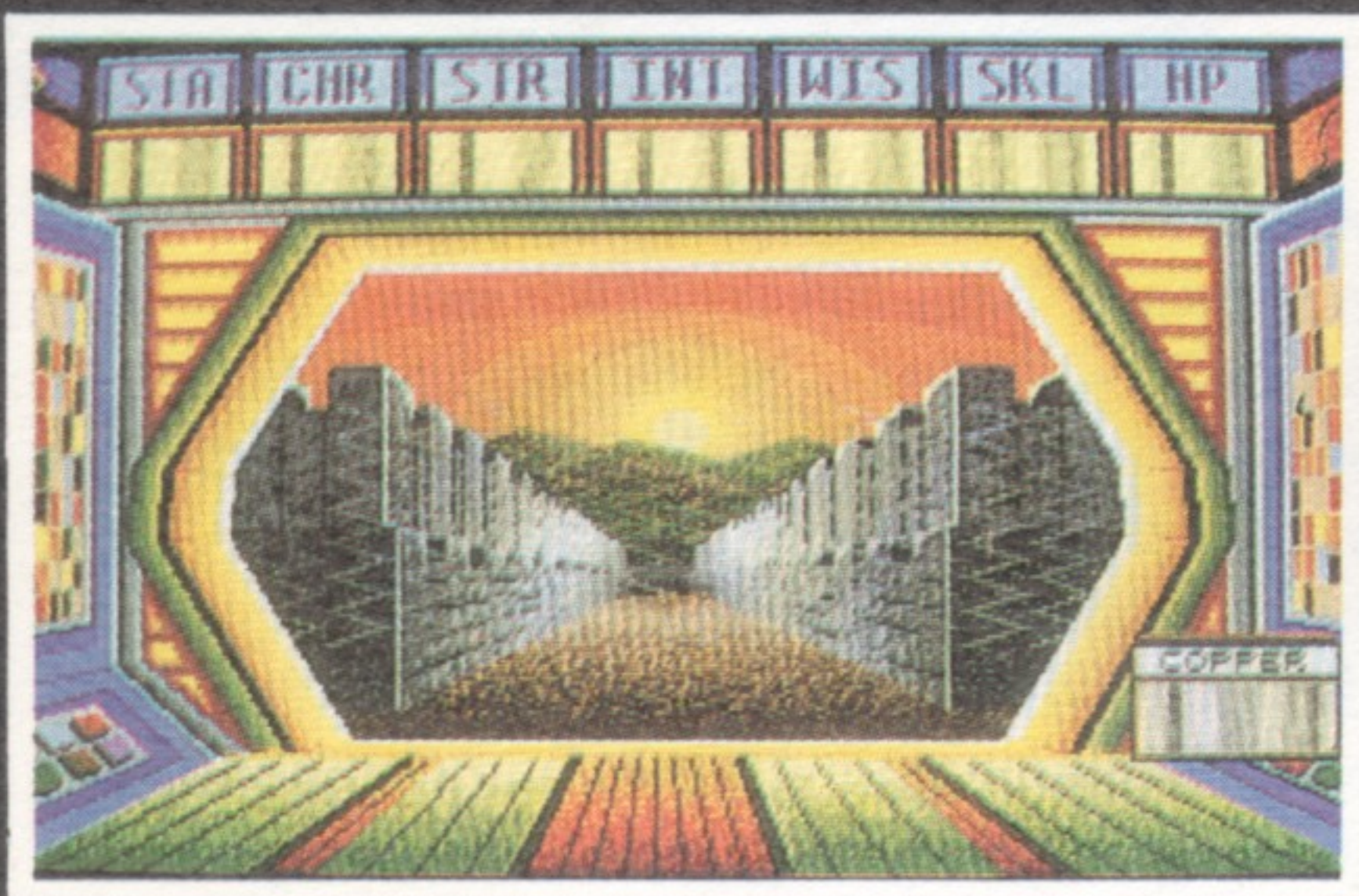
Conversione di un notissimo gioco apparso lo scorso anno, che ebbe notevole successo su diversi home computer. A suo tempo fu detto che Barbarian costituiva lo state of art in fatto di giochi di questa specie, quelli cioè nei quali le arti marziali la fanno da padrone. Noi non ci sentiamo di dire altrettanto di questa versione, non tanto perché sia stata fatta male ma perché, note le capacità grafiche di Amiga, lo state of art della macchina è senza dubbio ad un livello più alto. Questo non significa che il gioco non sia gradevole, tutt'altro: soprattutto quelli che hanno già lanciato colpi a destra ed a sinistra con le precedenti versioni, lo troveranno eccezionale. Il suono è senza dubbio all'altezza del nostro computer, poiché ogni colpo subito o inferto è accompagnato da un preciso rumore campionato, veramente notevole; per non parlare poi delle urla di vittoria o di sconfitta degli sfidanti. Potete combattere contro il computer oppure contro un vostro amico, situazione quest'ultima senz'altro molto più divertente. Controllerete TUTTI i possibili movimenti di un guerriero armato di sola spada e difeso unicamente da uno scudo metallico. Le mosse che potrete eseguire sono veramente moltissime, ad ogni combinazione del joystick ne corrisponde una. Si impiega un po' di tempo prima di impararle tutte, ma quando si è acquisita una certa padronanza dei movimenti, il divertimento è garantito. Sui lati dello schermo ci sono due indicatori a forma di serpente che si dibatteranno ogni qualvolta subirete un colpo: quando le sfere poste in alto saranno scomparse definitivamente, solo allora morirete ed un simpatico mostriciattolo vi trascinerà via. La mossa più spettacolare e più difficile da eseguire è quella che vi permette di tagliare, con un colpo solo, la testa del vostro avversario: se giocherete contro il computer, questi si accorgerà quasi sempre delle vostre intenzioni ed agirà di conseguenza. C'è un modo per fregarlo, però: basta distrarlo andando avanti ed indietro, farlo avvicinare, aspettare la sua mossa e colpirlo. Se avrete mantenuto una certa coordinazione, vedrete la testa del vostro avversario schizzare letteralmente via! ■

POWER STYX

Strettamente imparentato con il puzzle, anche se di diversa fattura, richiede in pratica di scoprire un'intera schermata selezionando di volta in volta delle zone chiuse, non importa quale forma esse abbiano. Inizialmente il pennellino può camminare soltanto lungo il bordo dello schermo; per cominciare ad addentrarsi nello spazio interno sarà necessario premere fuoco. Da questo punto in poi sorgono le difficoltà: se durante il percorso uno degli oggetti che popolano il paesaggio riuscirà a toccare la linea che state tracciando, perderete una vita. Se, al contrario, riuscirete a chiudere la linea e quindi a formare una superficie «recintata», magicamente apparirà una parte della schermata da scoprire. La prima è davvero molto bella, degna di Amiga e della sua risoluzione grafica; la seconda è migliore ancora e la terza... ve la lasciamo immaginare! Come capita quasi sempre con questo genere di giochi, a meno di non scoprire qualche trucchetto (e non è detto che nelle pagine di Tips & Tricks non appaia quanto prima), se si muore in un livello superiore al primo si deve necessariamente cominciare tutto da capo. Oltre che gli oggetti volanti, lungo il tracciato corrono una serie di mostriciattoli molto pericolosi che al minimo contatto vi faranno fuori. L'unico modo per eliminarli è quello di imprigionarli in un'area e chiuderla quando vi si troveranno dentro. A seconda del tipo di mostriciattolo vi verranno assegnati punteggi differenti; non dimenticate che, finché vi troverete sul tracciato, sarete al sicuro dagli oggetti volanti. ■

ADVENTURE

a cura di MARCO BROVELLI

Avventurati con noi nel dedalo spesso difficilmente districabile delle adventure. Se sei uno di quei pochi in gamba che son riusciti a risolverne, inviaci la prova della tua bravura; soluzioni e mappe verranno pubblicate in questo spazio dedicato ed il tuo nome citato.

Finalmente soddisfatta l'attesa di chi aspettava l'uscita della versione Amiga di «Alternate Reality», un'avventura originariamente disponibile per Commodore 64 da due anni circa. Non si tratta, a dir la verità, di un adventure vero e proprio ma piuttosto di un Tole Playing Game: un genere di gioco le cui origini vengono solitamente fatte risalire al celebre game da tavolo «Dungeons ad Dragons», nel quale i comandi non occorre siano inseriti nella consueta sintassi verbo-complemento e dove viene annessa grande importanza alle caratteristiche del personaggio del quale si vestono i panni nella vicenda (intelligenza, forza, abilità, fortuna). Queste peculiarità sono indicate a mo' di punteggio assegnato ad ogni categoria, modificabili all'inizio della partita in modo da permettere la creazione di personaggi sempre diversi e di variare quindi a piacere lo svolgimento e la difficoltà del gioco. «Alternate Reality» è ambientato in una città, esplorando la quale

si possono fare incontri, alcuni dei quali decisamente spiacevoli; gnomi. assassini, maghi, mercanti, messaggeri, gladiatori ed elfi sono solo alcuni dei molti personaggi che si incontrano per le vie della metropoli.

Per recuperare energie val la pena trascorrere la notte alla locanda, oppure acquistare qualche capo di vestiario o anche una bussola (molto utile!) in una delle botteghe; gli armaioli possono fornire nuove armi, precauzione necessaria se si vuole sopravvivere in questo ambiente ostile. Le banche sono disposte a prestare denaro; ci sono guaritori cui ricorrere e taverne dove recarsi per recuperare forza.

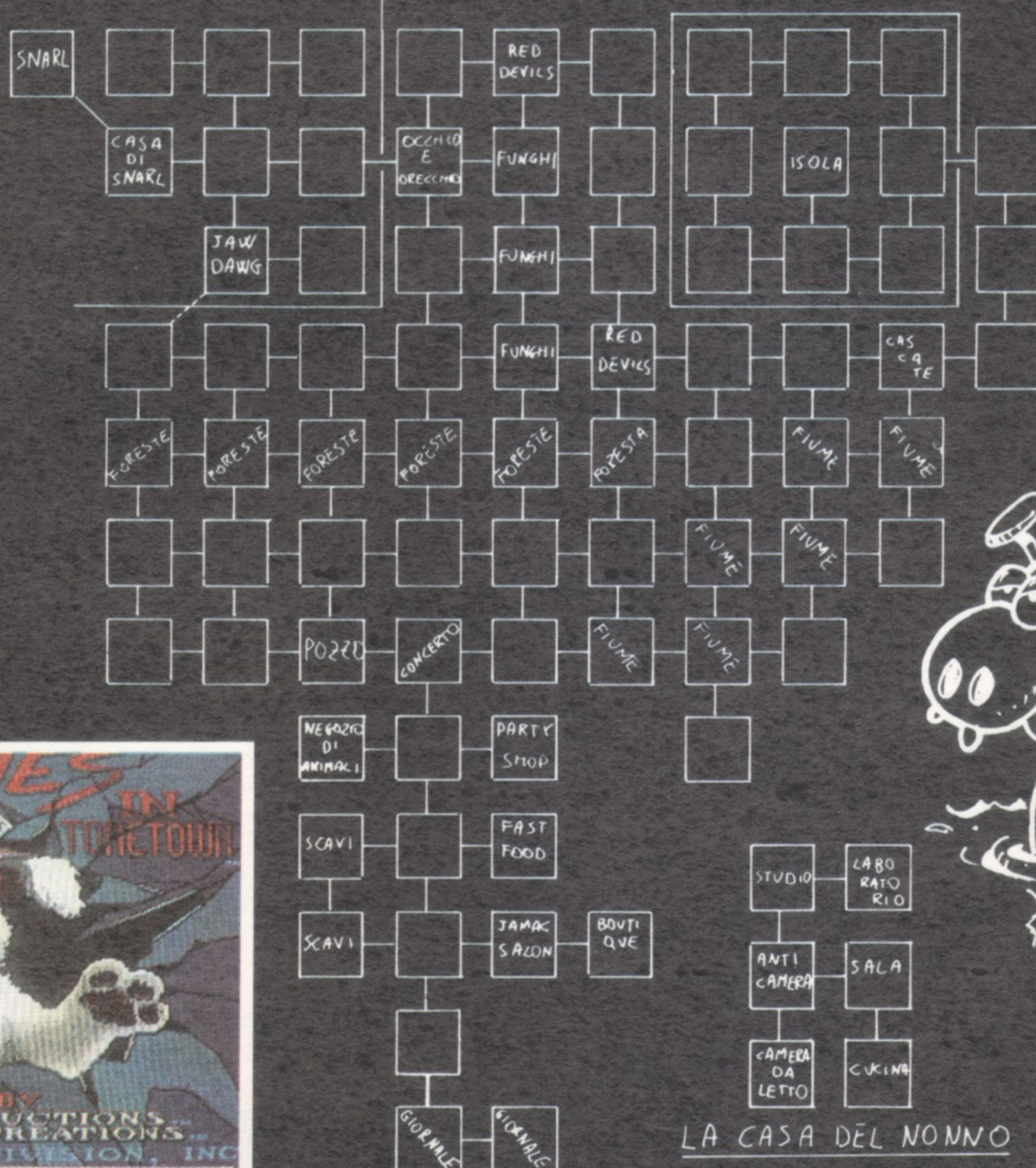
Può poi essere una buona idea quella di visitare una delle quattordici gilde dove, se riuscirete a farvi ammettere, aumenterete notevolmente la vostra esperienza ed il vostro punteggio.

La parte grafica è molto curata, ma è soprattutto dal punto di vista sonoro che il programma eccelle. Durante le vostre peregrinazioni per la città sarete infatti accompagnati da melodie di grande effetto, create con l'aiuto del celebre programma musicale «Sonix».

Non è possibile riuscire in poche righe a dare l'idea della complessità e dello spessore di «Alternate Reality», un gioco del quale non ci si stanca facilmente. Anche se gli appassionati di avventure di genere più tradizionale storceranno il naso perché rischiano di trovarlo un po' troppo statico per i loro gusti.

TASS TIMES IN TONE TOWN

la mappa

SNARL · CASA DI SNARL · JAW DAWG · RED DEVILS · OCCHIO E ORECCHIO · FUNGHI · ISOLA · CASCATE · FORESTE · FORESTA · FIUME · POZZO · CONCERTO · NEGOZIO DI ANIMALI · PARTY SHOP · SCAVI · FAST FOOD · JAMAC SALON · BOUTIQUE · GIORNALE · STUDIO · LABORATORIO · ANTI CAMERA · SALA · CAMERA DA LETTO · CUCINA

LA CASA DEL NONNO

Una tra le prime software-house a produrre avventure grafiche per Amiga è stata l'americana Activision, nota agli appassionati del genere per essere anche la Casa che distribuisce in tutto il mondo i prodotti Infocom. «Tass Times in Tone Town» è la loro terza avventura, dopo «Mindshadow» e «Borrowed Time», ad essere distribuita e, nonostante la sua prima apparizione sul mercato risalga a più di un anno fa, è tuttora molto popolare.

TASS TIME IN TONE TOWN

La trama del gioco è quantomeno insolita: preoccupati da un'inspiegabile mancanza di notizie da parte di vostro nonno, un bizzarro vecchietto che si diletta a costruire marchingegni strani, avete deciso di fare una visita al suo laboratorio per accertarvi del suo stato di salute. Il nonno, ahinoi, sembra scomparso; le uniche tracce rimaste di lui sono un libro zeppo di formule incomprensibili, una scatola piena di plettri per chitarra ed Ennio, il suo fedele cane, che continua ad agitarsi inspiegabilmente davanti ad una macchina dall'aspetto inusitato. Ben presto vi troverete trasportati a Tone Town, la città di un universo parallelo, nella quale il nonno è tenuto prigioniero da un minaccioso individuo chiamato Franklin Snarl, che ha le sembianze di un grosso serpente.

Nonostante l'apparenza tranquilla e pacifica dei suoi abitanti, Tone Town nasconde una quantità di sorprese, e bisogna riuscire a recuperare il nonno prima che qualcuno decida che la vostra presenza non è gradita...

Per conoscere tutti i segreti di Tone Town dovrete affidarvi al vostro intuito; e se proprio doveste rimanere a corto di risorse, potrete ricorrere alla soluzione completa del gioco, qui di seguito pubblicata. Ricordate che per completare «Tass Times in Tone Town» è necessario seguire alla lettera le istruzioni ed inserire tutti i comandi esattamente come li trovate scritti qui. Come misura precauzionale è consigliabile salvare spesso la posizione di gioco.

LA SOLUZIONE

S - EXAMINE COUNTER - EXAMINE JAR - GET KEY - N - W - UNLOCK DOOR - EXAMINE FISHBOWL - GET ALL - E - EXAMINE GENERATOR - THROW SWITCH - ENTER HOOP - N - LOOK TRENCH - GET JAR - DROP KEY - S - E - E - DYE HAIR - CHOOSE PINK - E - BUY JUMPSUIT - BUY HOOPLET - WEAR JUMPSUIT - WEAR HOOPLET - W - W - N - N - N - LOOK STELGAD - TALK TO STELGAD - S - S - S - S - S - TALK TO NUYU - GET CAMERA - E - TURN ON COMPUTER - TURN ON PRINTER - TYPE il tuo nome - YES - GET PASS - W - N - N - N - N - N - SHOW PASS TO STELGAD - USE CAMERA - DROP PASS - DROP CAMERA - TALK TO DAGLETS - GET ZAGTONE - LOOK PHOTO - WAIT - S - S - S - S - S - GIVE PHOTO TO NUYU - N - N - N - E - GET MITTS - WEAR MITTS - W - N - W - BUY BLOBPET - DROP PICKS - E - N - E - N - N - N - N - N - GET DEVIL - W - W - DROP JAR - N - W - DROP MITTS - GET CARD - E - S - E - S - E - GET MASK - GET BLACK - WEAR BLACK - W - N - W - GET JAR - D - W - W - N - N - N - U - E - N - UNLOCK GATE - S - W - D - S - S - S - E - E - U - DROP MASK - E - E - E - N - N - N - E - E - E - E - N - N - ENTER BOAT - W - S - W - N - N - W - INSERT CARD - PRESS BUTTON 5 - E - HIT CUFFS WITH ZAGTONE - W - PRESS BUTTON 1 - E - S - ENTER BOAT - S - W - W - W - GET MUSHROOM - THROW MUSHROOM AT EYE - OPEN GATE - W - W - W - N - SIC ENNIO ON SNARL - GIVE BOOK TO GRAMPS - OPEN JAR - PUSH SNARL IN HOOP.

La città di "Alternate Reality", popolata di gnomi, assassini, maghi, gladiatori, messaggeri...

Tanti programmi, recensioni, corsi, novità, rubriche ogni mese con trentaquattro mila lire di risparmio.

ABBONATI!

OCCASIONE SPECIALE, PREZZO STRACCIATO

Solo lire 120.000
per 11 fascicoli ed altrettanti dischetti
direttamente a casa tua.
(lire 55.000 per 5 fascicoli e 5 dischi)

Per abbonarti invia vaglia postale ordinario ad Arcadia srl, c.so Vitt. Emanuele 15, 20122 Milano. Oggi stesso, non perdere tempo!

FLAG DI STATO

3ª lezione

di LEONARDO FEI

Riprendiamo il programma pubblicato nella puntata precedente. Un modo diverso per scriverlo potrebbe essere:

```
start:
    MOVEQ   #$0F,D1        ;contatore del ciclo esterno
    LEA     $BFE001,A0     ;carica indirizzo in A0
ciclo:
    BSET    #1,(A0)        ;setta il bit 1
    BSR     ritardo        ;ciclo di ritardo
    BCLR    #1,(A0)        ;pulisce il bit 1
    BSR     ritardo        ;ciclo di ritardo
    DBRA    D1,ciclo       ;fine del ciclo esterno
    RTS                    ;fine
ritardo:
    MOVE.W#$FFFF,D0        ;contatore
pausa:
    DBRA    D0,pausa       ;ciclo di ritardo
    RTS                    ;fine subroutine
```

Troverete questo sorgente sul dischetto allegato a questo fascicolo, nella directory «corso_asm» con il nome «prog.1».

Innanzitutto notiamo che il programma è stato diviso in due parti, il corpo principale ed una routine esterna.

Quest'ultima si occupa semplicemente di produrre un certo

```
Tabella A
LEA    <arg1>,An

<arg1>:
   Dn              x   s8(An,Xn)
   An              x   xxxx.W
x  (An)            x   xxxx.L
   (An)+           x   s16(PC)
   -(An)           x   s8(PC,Xn)
x  s16(An)             #<dati>
```

Tabella A. Sintassi e modi di indirizzamento dell'istruzione LEA.

ritardo, in modo da rendere visibili i cambiamenti di stato della luminosità del LED. L'istruzione **LEA** (Load Effective Address) carica nel registro A0 l'indirizzo effettivo rappresentato dal suo operando, in questo caso $BFE001.

La sua sintassi ed i modi di indirizzamento sono contenuti nella tabella A.

Questa istruzione può essere utilizzata solo su registri Address e l'operando viene calcolato al momento dell'esecuzione. La cosa risulta indubbiamente vantaggiosa, come avremo modo di vedere in seguito, per scrivere programmi rilocabili. Un'altra istruzione che carica un indirizzo in un registro Address è **MOVEA**, ma non dobbiamo confondere le due. La sua sintassi è indicata nella tabella B. Mentre **LEA**, una volta calcolato l'operando, carica l'indirizzo ottenuto nel registro Address, **MOVEA** usa questo indirizzo per andare a prendere una word o longword che mette poi nello stesso registro Address.

In pratica, **LEA** carica un indirizzo effettivo, mentre **MOVEA** usa questo indirizzo per prendere il valore da mettere nel registro.

```
Tabella B
MOVEA.e   <arg1>,An

e = W, L

<arg1>:
x  Dn              x   s8(An,Xn)
x  An              x   xxxx.W
x  (An)            x   xxxx.L
x  (An)+           x   s16(PC)
x  -(An)           x   s8(PC,Xn)
x  s16(An)         x   #<dati>
```

Tabella B. Sintassi e modi di indirizzamento dell'istruzione MOVEA.

L'istruzione **BSET** (sintassi in tabella C) effettua due operazioni distinte: prima testa il bit specificato dal primo operando nella locazione di memoria specificata dal secondo operando, poi lo setta ad 1.

La prima operazione in questo caso non ci interessa.

L'istruzione **BSET #1,(A0)** setta quindi il bit 1 della locazione di memoria puntata da A0 ($BFE001).

L'effetto è equivalente a quello ottenuto con l'istruzione **MOVE.B #$02,(A0)**, solo che questa istruzione, oltre che settare ad uno il bit 1, pulisce gli altri 7 bit, mentre la **BSET** lascia gli altri bit inalterati.

Nel caso si utilizzi la prima notazione della sintassi l'operando, quello che stabilisce quale bit utilizzare, può essere compreso fra 0 e 7, mentre se si utilizza la seconda notazione, si può usare uno dei 32 bit del registro dati.

BSET #8, (A0)

non è legale, in quanto eccede la dimensione di un byte. Al suo posto si può scrivere:

MOVE.L #$100,D0
BSET D0,(A0)

Queste due istruzioni settano il nono bit a partire dall'indirizzo contenuto in A0.

(Nel nostro caso il registro $BFE001 è lungo un byte, ed utilizzando le due istruzioni appena enunciate andremmo a settare il primo bit della locazione successiva; $BFE002).

L'istruzione **BSR** (Branch SubRoutine) permette di passare momentaneamente il controllo della CPU ad una routine.

```
Tabella C
BSET.e   #<dati>,<arg2>
BSET.e   Dn,<arg2>
BCLR.e   #<dati>,<arg2>
BCLR.e   Dn,<arg2>

e = B, L

<arg2>:
x  Dn              x   s8(An,Xn)
   An              x   xxxx.W
x  (An)            x   xxxx.L
x  (An)+               s16(PC)
x  -(An)               s8(PC,Xn)
x  s16(An)             #<dati>
```

Tabella C. Sintassi e modi di indirizzamento delle istruzioni BSET e BCLR.

Prima viene salvato sullo Stack l'indirizzo della prossima istruzione (nel nostro caso **BCLR**), poi l'indirizzo specificato dall'operando viene messo nel Program Counter, quindi l'esecuzione del programma continua dalla nuova locazione. Per facilitarci la vita, l'assemblatore ci permette di utilizzare le «label» o etichette, che possiamo 'appiccicare' prima di certi parti del nostro programma.

Durante l'assemblamento o il linking queste etichette vengono tradotte in opportuni indirizzi ed il loro valore viene assegnato agli operandi delle istruzioni che ne fanno uso. Nel nostro caso, l'istruzione **BSR** permette unicamente l'indirizzamento relativo a Program Counter, quindi durante l'assemblamento l'etichetta «ritardo» viene tradotta in un opportuno spiazzamento rispetto al PC, e questo viene sostituito nella linea 'BSR ritardo'.

Analoga operazione viene effettuata quando l'assemblatore

```
Tabella D

Istruzione MOVE:

N  Settato secondo il
   risultato dell'operazione
Z  Settato se il risultato e'
   zero, altrimenti pulito
V  Sempre pulito
C  Sempre pulito
X  Non influenzato

Istruzione LEA:

N  Non influenzato
Z  Non influenzato
V  Non influenzato
C  Non influenzato
X  Non influenzato

Istruzione MOVEA:

N  Non influenzato
Z  Non influenzato
V  Non influenzato
C  Non influenzato
X  Non influenzato

Istruzione BSET e BCLR:

N  Non influenzato
Z  Settato se il risultato
   e' zero, altrimenti pulito
V  Non influenzato
C  Non influenzato
X  Non influenzato
```

Tabella D. Effetti provocati sui flag di stato da alcune istruzioni.

incontra la seconda istruzione 'BSR ritardo', ma questa volta lo spiazzamento è diverso, in quanto ci troviamo più vicini all'inizio della routine 'ritardo:'.

Dal momento che l'istruzione BSR utilizza l'indirizzamento relativo a PC, lo spiazzamento è specificato da 8 bit nel caso di un salto compreso fra —128 e +127 bytes, oppure da 16 bit per salti compresi fra —32768 e +32767 bytes.

Per eseguire salti più lunghi, nell'area dei 16 Megabytes disponibili, bisogna utilizzare l'istruzione JSR, della quale parleremo in seguito.

Dopo il ciclo di ritardo, generato con il metodo che abbiamo già visto, la CPU incontra l'istruzione RTS (Return Subroutine), riprende l'indirizzo che aveva salvato dallo Stack e lo mette nel PC. L'esecuzione del programma riprende quindi dall'istruzione **BCLR**, e si dice che il programma è ritornato dalla

```
Amiga_Byte 3:Corso_Asm
2> dir
  .info                              prog.1
  prog.1.info
2>
2> type prog.1
start:

   MOVEQ    #$0F,D1            ;contatore del ciclo esterno
   LEA      $BFE001,A0         ;carica indirizzo in A0

ciclo:
   BSET     #1,(A0)            ;setta il bit 1
   BSR      ritardo            ;ciclo di ritardo
   BCLR     #1,(A0)            ;pulisce il bit 1
   BSR      ritardo            ;ciclo di ritardo
   DBRA     D1,ciclo           ;fine del ciclo esterno
   RTS                         ;fine

ritardo:
   MOVE.W   #$FFFF,D0          ;contatore
pausa:
   DBRA     D0,pausa           ;ciclo di ritardo
   RTS                         ;fine subroutine

2>
```

Stampa del programmino in assembler contenuto nel dischetto allegato a questo numero di Amiga Byte, ottenuta con un semplice Type del file.

subroutine. L'istruzione **BCLR** fà esattamente l'opposto della sorella **BSET**: prima testa, poi pulisce il bit specificato. **DBRA** chiude il ciclo ed RTS termina il programma e ritorna il controllo alla CLI.

IL TEST SUL BIT

Vediamo adesso cosa significa esattamente la parola "testare" che abbiamo incontrato più di una volta.

Prendiamo, per esempio, l'istruzione **BSET**. Questa, prima di settare il bit, lo «testa» (dall'inglese «test» = provare), va cioè a vedere cosa contiene, ed influenza lo Status Register in relazione a quello che ha trovato. Le istruzioni **BSET** e **BCLR** influenzano solo il flag Z, quello che segnala se il risultato di una certa operazione è zero. In questo caso, se prima dell'istruzione **BSET** o **BCLR** il bit in questione era a zero, il flag Z è settato (1), indipendentemente dalla condizione in cui esso si troverà alla fine.

Se invece il bit era settato, il flag Z viene pulito. Gli altri flag non sono influenzati dalle istruzioni **BSET** e **BCLR**. A questo punto è necessario rivedere le tabelle con la sintassi dei comandi, per includere anche questo tipo di informazione: quali sono i flag dello SR (detti anche «status flag» = flag di stato) che vengono influenzati da ciascuna operazione.

Nella tabella D sono elencati gli effetti sui flag di stato delle istruzioni per le quali abbiamo già visto la sintassi. La dicitura 'Settato secondo il risultato dell'operazione' ha vari significati, a seconda del flag in questione.

Il flag N, per esempio, viene settato quando il bit più alto dell'operando è settato, in quanto questo bit convenzionalmente rappresenta il segno ed il flag N vuole dire appunto Negativo.

In pratica, un numero rappresentato da un byte può essere considerato senza segno, nel qual caso assume un valore compreso fra 0 e 255. Se prendiamo in considerazione il segno, per convenzione rappresentato dal bit più alto, detto anche il bit più significativo, rimangono 7 bit per rappresentare il valore assoluto. Il byte può quindi assumere un valore compreso fra —128 (tutti gli 8 bit ad 1) e +127 (i primi 7 bit a 1 e l'ottavo, il segno, a 0).

Questo discorso si ripete per le word, nelle quali il bit che rappresenta il segno è il sedicesimo (bit 15), e per le longword

(bit 31 = segno). Facciamo alcuni esempi:

```
MOVE.B #$75,D0
;carica D0 con il byte $75.
```

Il flag N viene pulito, in quanto il bit più alto (in questo caso l'ottavo) non è settato.

```
MOVE.B #$82,D0
carica D0 con il byte $82.
```

Il flag N viene settato perché il bit 7 è settato ($82, in binario è %10000010).

```
MOVE.W #$82,D0
;carica D0 con la word $0082.
```

Il flag N viene pulito, perché il bit più alto (in questo caso il sedicesimo) non è settato.

In tutti questi casi il flag N viene pulito, in quanto l'operando è diverso da zero.

Spendiamo adesso una parola di avvertimento circa le confusioni che potrebbero derivare dalle istruzioni **MOVE** e **MOVEA**.

Alcuni assemblatori, il Seka ad esempio, non permettono l'utilizzazione del simbolo mnemonico **MOVEA**. In questi casi si usa **MOVE**. Ci penserà l'assemblatore a tradurre tutto nei codici appropriati. Quello cui bisogna fare attenzione però, è che, mentre l'istruzione **MOVE** influenza i flag di status, l'istruzione **MOVEA** non li modifica.

L'EFFETTO SUL FLAG

Scrivendo un sorgente per il Seka, utilizzeremo sempre **MOVE**, ma mentre l'istruzione **MOVE #0,D0** setta il flag Z, **MOVE #0,A0 (MOVEA)** non ha effetto su alcun flag.

Un'altra causa di frequenti errori è data dall'impiego di **MOVE** e **LEA**:

```
MOVE.L Kickstart,A0
Kickstart: DC.L $436A00
```

carica in A0 la longword $436A00, mentre

```
LEA Kickstart,A0
MOVE.L (A0),D0
Kickstart: DC.L $436A00
```

carica in A0 il puntatore alla longword $436A00.

Successivamente questo puntatore viene utilizzato per caricare la longword in D0.

A DESTRA DELL'ISTRUZIONE

Un'ultima parola sull'istruzione appena introdotta: DC.e

Questa non è una istruzione della CPU, ma una direttiva che viene interpretata dall'assemblatore.

Serve ad immettere determinati valori nel programma.

La sintassi è **DC.e #<dati> con e = B, W, L.**

Per esempio:

```
DC.B $34,$21,$33,"Ciao"!,0
```

crea in memoria una sequenza di bytes come specificato dalla parte a destra dell'istruzione.

Nel caso della stringa «Ciao!», questa viene tradotta in una sequenza di corrispondenti codici ASCII. ■

SE QUESTO FASCICOLO TI È PIACIUTO SCRIVICELO

...ma anche se non ti è piaciuto, naturalmente. Ci interessa molto il tuo parere perché può aiutarci a darti proprio quello che vuoi. Rispondi per cortesia a queste domande. Grazie.

Quanti anni hai? ....

Se studi, che studi fai? ....

Se lavori, che lavoro fai? ....

Quale Amiga possiedi? ☐ 500 ☐ 1000 ☐ 2000

Ti è piaciuto questo fascicolo? ☐ sì ☐ no

Cosa ti è piaciuto di più? ....

Hai dei suggerimenti? Quali? ....

NOME E COGNOME

INDIRIZZO TEL.

CITTÀ C.A.P. PROV.

Completa con il tuo indirizzo solo se vuoi e spedisci questo tagliando o una fotocopia ad AMIGA BYTE, Arcadia c.so Vitt. Emanuele 15, 20122 Milano.

Potenziamo la stampante

Superiamo i limiti della nostra stampante ampliandone incredibilmente le possibilità di utilizzo con l'aiuto di tre programmi versatilissimi.

di ALBERTO RANZANI

La stampante è divenuta ormai, con l'evoluzione dell'informatica home, strumento indispensabile se si vogliono sfruttare adeguatamente tutte le possibilità offerte dalle nuove, potenti macchine a nostra disposizione. Un tempo costose e complicate, le stampanti sono oggi economiche, compatte, semplici da usare ed offrono buone prestazioni.

Sarebbe superfluo sottolineare l'importanza del poter mettere su carta i documenti scritti con il word processor, i calcoli eseguiti con lo spreadsheet o, nel caso di applicazioni più creative, la comodità di avere una copia stampata dei propri disegni, tecnici o artistici: tutti compiti che una buona stampante economica ad ottanta colonne, il tipo più diffuso, svolge egregiamente.

CRESCONO LE ESIGENZE

Ma se le nostre esigenze aumentano? Se ci interessasse non esser costretti a limitare le dimensioni di hardcopy del disegno fatto con il De Luxe Paint, oppure, se abbiamo creato uno spreadsheet più largo di ottanta colonne o, ancora, se l'ultimo capolavoro di progettazione eseguito con l'Aegis Draw meriterebbe una stampa più accurata di quella eseguibile normalmente ed ovviamente non abbiamo il plotter? Ecco allora venirci in aiuto tre utilissimi programmi di stampa: Plot to Printer, Flipside e Huge-Print.

Plot to Printer, della americana Hi-Tech Graphics, trasforma la stampante in un quasi-plotter in formato A4; quasi, perché del plotter non utilizza i colori (anche qualora li abbia a disposizione in una stampante a colori). La tecnica usata è molto interessante: mentre l'hardcopy di un disegno tecnico su stampante avviene semplicemente trasferendo su quest'ultima la pagina grafica che appare sul monitor (con le conseguenti deformazioni), il disegno su plotter avviene inviando al plotter stesso delle stringhe di comando che gli indicano coordinate, spostamenti e direzioni relative; queste stringhe seguono géneralmente il formato HPGL della Hewlett-Packard che è divenuto, per i plotter, uno standard paragonabile a quello Epson per le stampanti.

Poiché poi normalmente il disegno su plotter è piuttosto lento, i programmi CAD consentono di creare su disco un file contenente i comandi che vengono poi mandati al plotter come task di background (l'Amiga, si sa, è multitasking) mentre il programma si rende di nuovo disponibile all'utente. Plot to Printer si occupa quindi di convertire questo file in un altro, che contiene una pagina di stampa in super alta densità; funziona quindi con tutti i CAD in grado di creare files HPGL su disco (Aegis Draw, Aegis Draw +, X-CAD...).

Contenuto del dischetto TurboPrint aperto da Workbench: nel file Readme c'è una interessante tabella per la compatibilità dei driver per stampanti.

STAMPANTI EPSON-COMPATIBILI

La versione del programma in nostro possesso genera un file stampabile con stampanti Epson-compatibili, ma tutte le stampanti grafiche che supportano la super high dens sono utilizzabili (quindi anche le IBM-compatibili).

Sul dischetto è presente un manuale in linea, ricco di esempi ma non sempre preciso; un vero peccato, causa il quale l'utilizzo del programma finisce con l'essere un pochino macchinoso. Ma procediamo. Muniamoci di due dischetti vuoti formattati, quindi carichiamo il CAD e salviamo su uno dei due dischi il file di plot che vogliamo convertire: il file deve chiamarsi PLOT; se così non fosse, bisognerà operare un Rename. Un consiglio: non salvate il disegno con l'opzione GRIGLIA attivata, perché questa verrebbe fedelmente

Esempio di piccolo poster creato con «Huge Print», utilizzando una schermata IFF. Le dimensioni reali sono di circa 40x35 cm; una maggior cura nell'assemblaggio delle strisce avrebbe certo migliorato l'effetto finale!

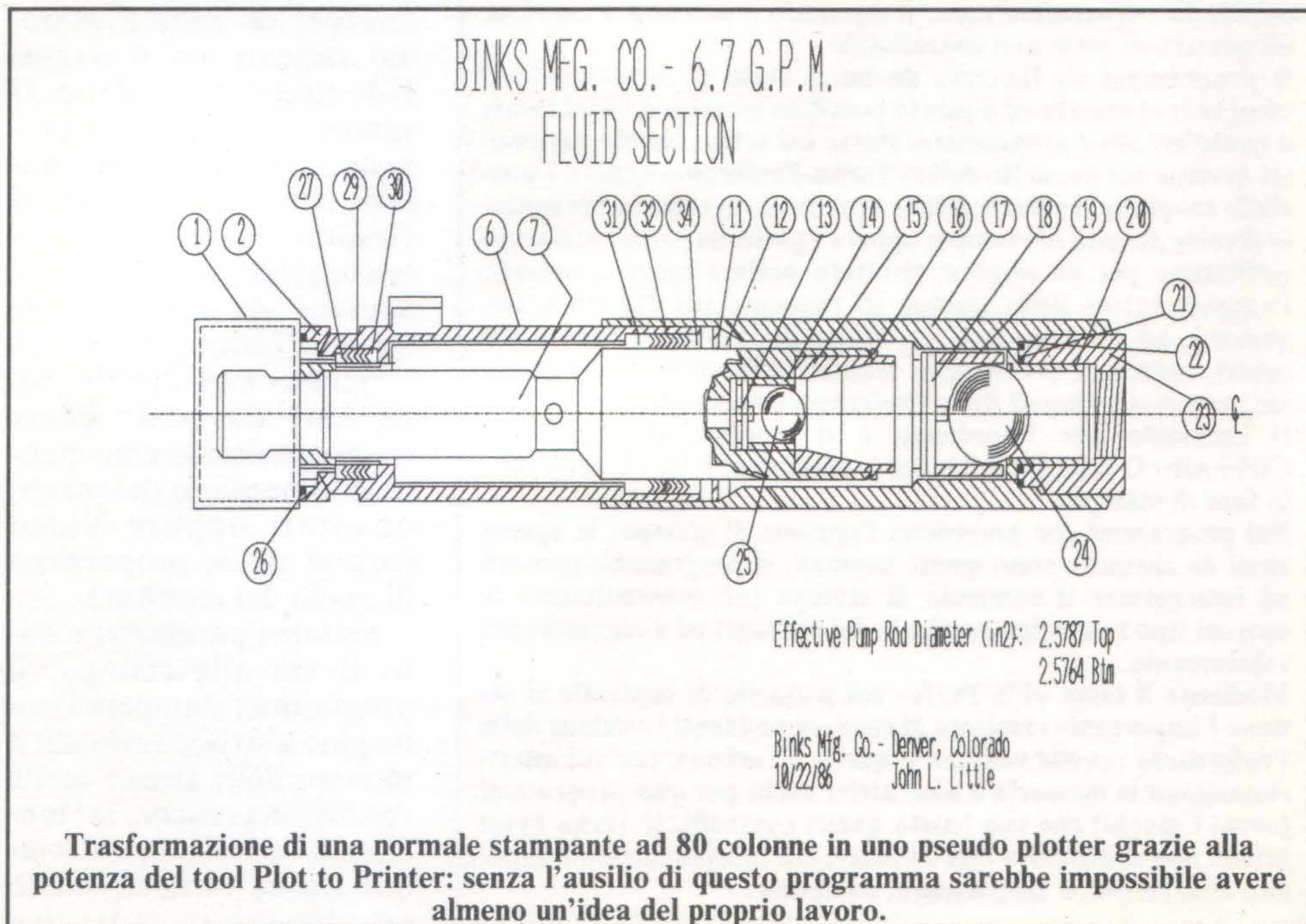
riportata. Mettiamo ora un disco Cli nel df0:, il disco Plot to Printer nel df1: (il secondo drive è indispensabile), aumentiamo lo stack portandolo a 200.000 (per disegni non eccessivamente complessi basta 150.000), andiamo in df1: con cd e battiamo «plot.epsh»; da questo momento il programma si occuperà di indicarci come proseguire e ci terrà aggiornati sulla conversione. Quest'ultima durerà parecchio, in media 20 minuti circa; il tempo dipenderà dalla complessità del disegno.

MODIFICHIAMO I PARAMETRI

Possiamo indicare al programma cinque parametri supplementari, da inserire dopo il plot.epsh, che rappresentano: la rotazione del disegno di 90 gradi, la scala sull'asse x, quella sull'asse y, lo spostamento del disegno rispetto a questi assi. È inoltre possibile creare stampe a densità inferiore (plot.epsq o plot.epsd, cioè quad o double) o far sì che il procedimento di conversione avvenga in ram (usando Qplot...), il che rende il tutto un po' più veloce ma necessita di almeno 1,5 Mb di memoria.

La rotazione è utilizzata per stampare il disegno in verticale o in orizzontale; la scala, invece, perché il programma calcola le coordinate del disegno e le modifica per ottenere per default una grandezza a tutta pagina (dimensione 1 sugli assi x ed y). Il disegno può venire, di conseguenza, sfalsato nelle proporzioni. Aggiustando la scala su uno degli assi, o su entrambi, si risolve il problema: ad esempio, se un quadrato si trasforma in un rettangolo il cui rapporto tra lato maggiore e lato minore è 2:1, basterà specificare una scala sull'asse del lato maggiore di 0.5. Variando lo spostamento sugli assi x ed y si può centrare il disegno o farlo apparire ovunque sulla pagina.

Terminata la conversione dei files con un salutare: «file creation was successful», è consigliabile inserire di nuovo il disco Cli nel df0:, quindi il disco programma nel df1: e copiare in ram: il file «print.epsh» (o print.epsd...a seconda della densità del file plot). Inserite poi nel df1: il disco con il file convertito e battete: «cd df1: — stack 150000 — ram: print.epsh»: la stampante si metterà in funzione. Se fosse collegata tramite l'interfaccia seriale, al comando si dovrà far eseguire la parola chiave SER. Tutta

Trasformazione di una normale stampante ad 80 colonne in uno pseudo plotter grazie alla potenza del tool Plot to Printer: senza l'ausilio di questo programma sarebbe impossibile avere almeno un'idea del proprio lavoro.

TURBO PRINT

Il Turbo Print (Prodotto da IrseeSoft, Grunterstrasse 6, D-8951 Irsee, Germania Ovest e distribuito da Logitek, via Golgi 60, Milano, 02/26.66.274) segue il filone inaugurato dal famoso Grabbit; il programma permette cioè, una volta installato in memoria, di eseguire una rapida hardcopy dello schermo corrente del computer.
In realtà esso è però più orientato alla stampa vera e propria:

Hardcopy della schermata delle TurboPreferences ottenuta con lo stesso Turbocopy: la densità usata è di 180 per 180 dpi (punti per pollice).

non è possibile infatti salvare la schermata su disco od in ram per un uso futuro, ma la sua maggior potenza è derivata dal fatto che questo programma «scavalca» i driver delle stampanti presenti nelle Preferences, utilizzando i propri che sono molto più efficienti. In questo modo la fase di stampa avviene con una velocità che arriva ad essere superiore di dieci volte a quella normale. La velocizzazione è tanto più evidente quanto più complicati sono i drivers e le pagine da stampare (per esempio una pagina a 4096 colori ed una stampante a 24 aghi).
Correda questo software un manualetto di 20 pagine che comprende la registration card; il dischetto è protetto e contiene un numero di serie non cancellabile.
Il programma va lanciato da boot; dopo il caricamento il computer si resetta ed è perciò possibile inserire il Workbench o qualsiasi altro programma. Parte del settaggio dei parametri avviene nel pannello delle «Turbo-Preferenze», dove è possibile scegliere il printer driver e, se la stampante lo permette, le diverse densità di stampa; oppure i patterns per le retinature (utilissime per un miglior risultato nella stampa a colori), l'«aspect ratio» della stampa (il rapporto tra altezza e larghezza), ed altro ancora. La scelta del bianco e nero o dei colori, invece, o quella della stampa orizzontale o verticale, etc, vanno selezionati dalle Preferences normali.
Il comando per l'hardcopy è il classico (Ctrl+Alt+P; Ctrl+Alt+O stampa in bianco e nero; Ctrl+Alt+C abortisce la fase di stampa.
Nei programmi che prevedono l'opzione di stampa, le operazioni da eseguire sono quelle normali: il programma penserà ad intercettare il comando di stampa (ed eventualmente le opzioni tipo la stampa verticale del DPaint) ed a eseguirlo più velocemente.
Mediante il tasto «Fix Prefs» nel pannello di controllo si ottiene l'importante risultato di rendere residenti i settings delle Preferences (quelle normali e quelle «Turbo»), per cui questi rimangono in memoria e sono attivi anche per quei programmi (come i giochi) che non hanno questi controlli. Il Turbo Print infatti non si disattiva con un reset, ma facendo girare l'apposito «NoTurbo» o spegnendo il computer.

questa procedura può forse spaventare, a volte è addirittura necessario fare diverse prove per aggiustare correttamente le dimensioni del disegno ma, «fattaci la mano», i risultati non tarderanno ad arrivare e saranno notevoli. Con un nastro nuovo ed una carta di buona qualità i disegni saranno praticamente perfetti, adatti anche ad un uso professionale. D'altra parte, l'alternativa più economica costa circa due milioni di lire!

ARRIVA LO HUGEPRINT

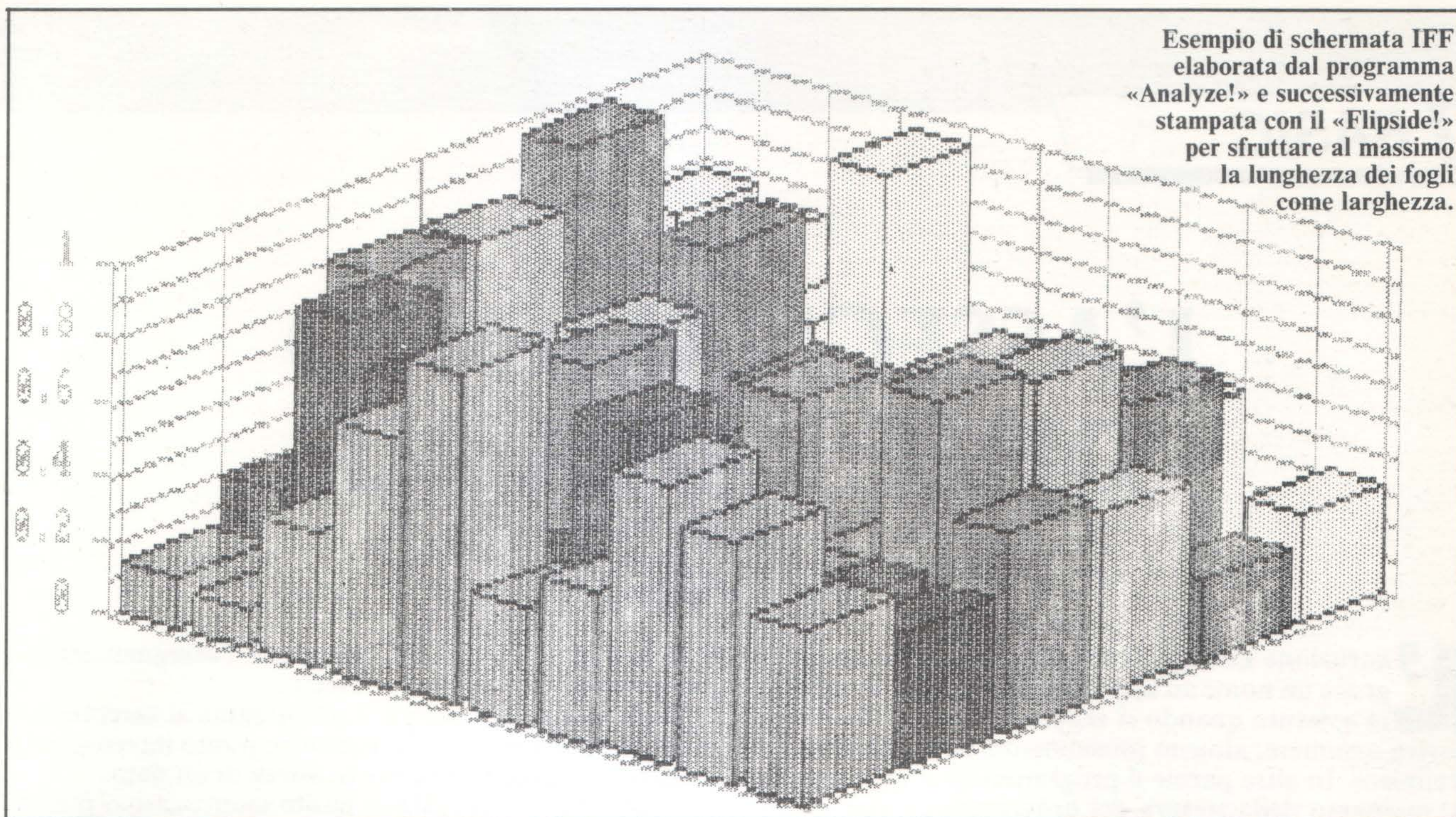
Il programma HugePrint, della americana Hugh's Software Ranch, permette di creare posters di diverse dimensioni partendo da una qualsiasi immagine nello standard IFF; può essere utilizzato con tutte le stampanti previste nel workbench ed ha incluse le preferenze.

Una volta caricatolo, i classici menù a tendina consentono di compiere la prima operazione, di scegliere cioè «help+unjam» per rendere più «umano» il caotico schermo di presentazione; di seguito potremo caricare un'immagine IFF (quelle di DPaint II vanno «depacked» caricandole e, di seguito, salvandole con il Graphicraft, il Graphics Studio, il Pixmate etc.). Sceglieremo poi il numero di «fette» nelle quali l'immagine verrà «tagliata» e quante di queste fette dovranno essere stampate; decideremo quindi le dimensioni dei pixels, quadrati oppure Video (con le stesse proporzioni di quelli del monitor).

Settati i parametri e dato il via alla stampa, il programma stamperà l'immagine a strisce verticali; il numero delle strisce scelte condiziona anche la lunghezza delle stampate (cinque strisce verticali danno una lunghezza indicativa di circa 28 cm.; la larghezza dell'immagine derivante dall'unione delle cinque strisce è di circa 45 cm.).

Una volta terminata la stampa, operazione piuttosto lunga, il nostro compito diviene alquanto arduo: bisogna munirsi di forbici, colla e tanta pazienza. Ritagliando le strisce ed incollandole una di fianco all'altra si comporrà il poster. In questa fase è necessaria molta precisione nel seguire i contorni delle stampate, ed è importante non sbagliare l'ordine di accoppiamento; se poi si è scelto il massimo formato (16 strisce, che danno un'altezza di più di due metri), il tutto diventa veramente impegnativo. Esattamente il contrario dell'uso del programma, molto immediato. I risultati sono però notevoli se il lavoro di giuntura viene fatto bene: se siete già dei maghi con il taglia ed incolla dei word processors, la cosa non dovrebbe spaventarvi!

ECCO IL FLIPSIDE!

Il Flipside!, creato dalla Microsystems inc. (quella di Analyze! Scribble! Online!, con i quali è compatibile), permette invece di trattare un'immagine in formato IFF od un testo in formato ASCII in modo da poterli stampare in verticale, utilizzando la lunghezza dei fogli come fosse la larghezza. Il programma lavora in multitasking e va lanciato come task 2 da Cli o da Workbench; una volta creato o trovato il file da modificare con il programma di grafica, con il wp o con lo spreadsheet, si richiama il flipside; ciò è necessario perché il programma non consente di vedere il file che deve essere stampato. Se si dovesse rendere necessario stampare il file di un programma che non ammette il

Esempio di schermata IFF elaborata dal programma «Analyze!» e successivamente stampata con il «Flipside!» per sfruttare al massimo la lunghezza dei fogli come larghezza.

multitasking sarà necessario salvarlo e ricordarne esattamente il nome, in quanto poi lo si dovrà «dare» a Flipside, che lo cercherà e lo stamperà.

Lo schermo del Flipside appare come un requester riposizionabile ed una barra dei menù, tra i quali quello che permette anche l'archiviazione dei files da stampare qualora l'operazione debba essere ripetuta più volte nel tempo.

Il requester mostra i gadgets riguardanti le dimensioni di stampa, i margini, il font da utilizzare, le linee per l'eventuale «incollatura» (i fogli si possono incollare sovrapponendoli). Le dimensioni della stampa possono essere variate a piacere, mediante il Form set, ed è presente un comodo «status» che indica quali sono le dimensioni massime (in dots) dei fogli della stampante settata nelle preferenze.

È poi possibile modificare il font con il quale il file verrà stampato; oltre che poter caricare i fonts esterni, se ne possono variare le dimensioni in tre formati. Tutti i margini possono essere settati: superiore, inferiore e laterali; le «glue lines» scelte vengono lasciate bianche affinché si possano incollare.

Va sottolineato che il programma non accetta dati incompatibili con le dimensioni disponibili; inoltre, settando le dimensioni del Form (cioè della pagina), esso ci comunicherà automaticamente le linee e le colonne che potremo stampare. Il programma può essere utilizzato per comprimere o modificare la stampa anche in senso orizzontale. La qualità di stampa è più che accettabile, la velocità invece lascia un poco a desiderare.

L'uso del Flipside! è abbastanza immediato; è necessario però, anche in questo caso, fare delle prove per abituarsi ai parametri utilizzati, il che non è per nulla complicato. Di contro, avremo a disposizione uno strumento che aumenterà sensibilmente la versatilità della nostra stampante.

Le prestazioni di una normale stampante possono perciò essere notevolmente migliorate con l'impiego di questi programmi, per mezzo dei quali è possibile produrre stampe di qualità solitamente ottenibile solo con apparecchiature dal costo molto elevato.

Il loro uso è perciò consigliato soprattutto per applicazioni amatoriali evolute, quando non si hanno a disposizione grandi somme di denaro.

■

DOVE I PROGRAMMI

HUGEPRINT - «Hughs Software Ranch», 50 East end Avenue, New York, NY 10028. Prezzo: 48 dollari.
FLIPSIDE - «Microsystems Inc.», 4301-18 Oak Circle, Boca Raton FL 33431. Prezzo: 30 dollari.
PLOT TO PRINTER - «Hi-Tech Graphics», P.O. Box 446, Tallmadoe, Ohio 44278.

L'ASSEGNAMENTO DELLE VARIABILI

3ª parte

di LUCA BRIGATTI

L'istruzione Let che, come si è visto, serve per assegnare un nome ad una variabile, può venir utilizzata soltanto quando si sappia a priori quale valore dovrà assumere, almeno inizialmente, la variabile che si definisce. In altre parole il programmatore deve sapere, al momento della stesura del programma, quale valore deve assumere quella determinata variabile.

Nel programma didattico presentato la volta scorsa, il prezzo iniziale dei capi, l'ammontare dell'Iva e quello dello sconto venivano definiti in partenza; il programma può essere così utilizzato soltanto a partire da dati sempre uguali, il che lo rende pressoché inutile in pratica.

Molto meglio sarebbe poter creare un programma che esegua gli stessi calcoli ma su variabili definite di volta in volta a seconda dell'esigenza. Ciò è possibile grazie all'istruzione INPUT (letteralmente «Metti dentro»).

Quando un programma incontra tale istruzione si ferma ed attende, per proseguire, che l'utente inserisca un dato, numerico o alfanumerico secondo la necessità.

Tale dato verrà memorizzato in una variabile che dovrà, ovviamente, essere di tipo coerente al dato stesso.

Alcuni esempi:

Input a
Input Prezzo%
Input Distanza#
Input Giorno$
Input Giorno$,Mese$,Anno%

Si notino il diverso tipo di variabili usate e la possibilità di includere in una sola istruzione l'assegnamento di più variabili, anche di tipo diverso.

```
BASIC
Commodore Amiga BASIC
Version 1.2
Created  6 Oct 1986
Copyright (c) 1985, 1986
 by Microsoft Corp.
 487272 Bytes free in System
 25000 Bytes free in BASIC

LIST
  INPUT a
  INPUT Prezzo%
  INPUT distanza#
  INPUT Giorno$
  INPUT Giorno$,Mese$,Anno%

REM **** Notare il diverso tipo ****
REM **** di variabili usate ********
```

A seconda del tipo di dato da immettere, è necessario scegliere una precisa variabile, riconoscibile dal suffisso.

Negli esempi precedenti il programma si sarebbe fermato ed avrebbe fatto comparire un punto interrogativo «?» per farci capire di essere in attesa di un dato.

Al posto o in aggiunta al punto interrogativo possiamo far sì che venga scritto un messaggio qualsiasi:

Input "Scrivi il Prezzo unitario",Prezzo
Input "Quanto Pesa";Peso

Il messaggio, come si vede, deve essere racchiuso fra virgolette e separato dalla variabile per mezzo di un delimitatore che può essere la virgola «,» o il punto e virgola «;». Nel primo caso il sistema NON aggiungerà il punto interrogativo, nel secondo sì.

Tre input di variabili numeriche racchiusi nella caratteristica finestra List del basic con l'opzione STEP del menu Run selezionata.

Tornando al nostro esempio, potremmo sostituire le istruzioni Let con i corrispettivi Input:

Input "Scrivi il Prezzo Iniziale",Prezzo
Input "Valore dell'IVA";IVA
Input "Ammontare dello sconto";Sconto

Provate e vi renderete conto che nella nuova versione il programma risulta, visto il tipo di calcolo che è comunque fisso, più versatile ed in grado di adattarsi a diversi casi; in una parola, più *flessibile*. La flessibilità, qualsiasi sia la funzione del programma, rende migliore il programma stesso e va sempre ricercata.

Se, al momento di inserire il dato, qualcosa non torna (ad esempio il dato inserito non è coerente con il tipo di

variabile specificata) si otterrà il messaggio di errore «Redo from start» ovvero «Ripeti dall'inizio»; con ciò il sistema ci invita a ripetere l'inserimento dalla prima variabile specificata in quella istruzione Input. Usare Input come comando, cioè in esecuzione diretta, non ha senso.

GLI OPERATORI ARITMETICI

Oltre che gli operatori ecologici, gli operatori turistici e gli operatori di borsa, esistono anche gli Operatori Aritmetici. Alcuni di essi abbiamo già avuto il piacere di incontrarli, sono quei piccoli segnetti che indicano le normali operazioni, i «conticini» di vecchia memoria. Cominciamo con somma e sottrazione, rappresentati dai simboli (guardacaso) «+» e «—».

Ci sono poi la moltiplicazione e la divisione, rappresentati dai simboli (qui non indovinate) «*» e «/».

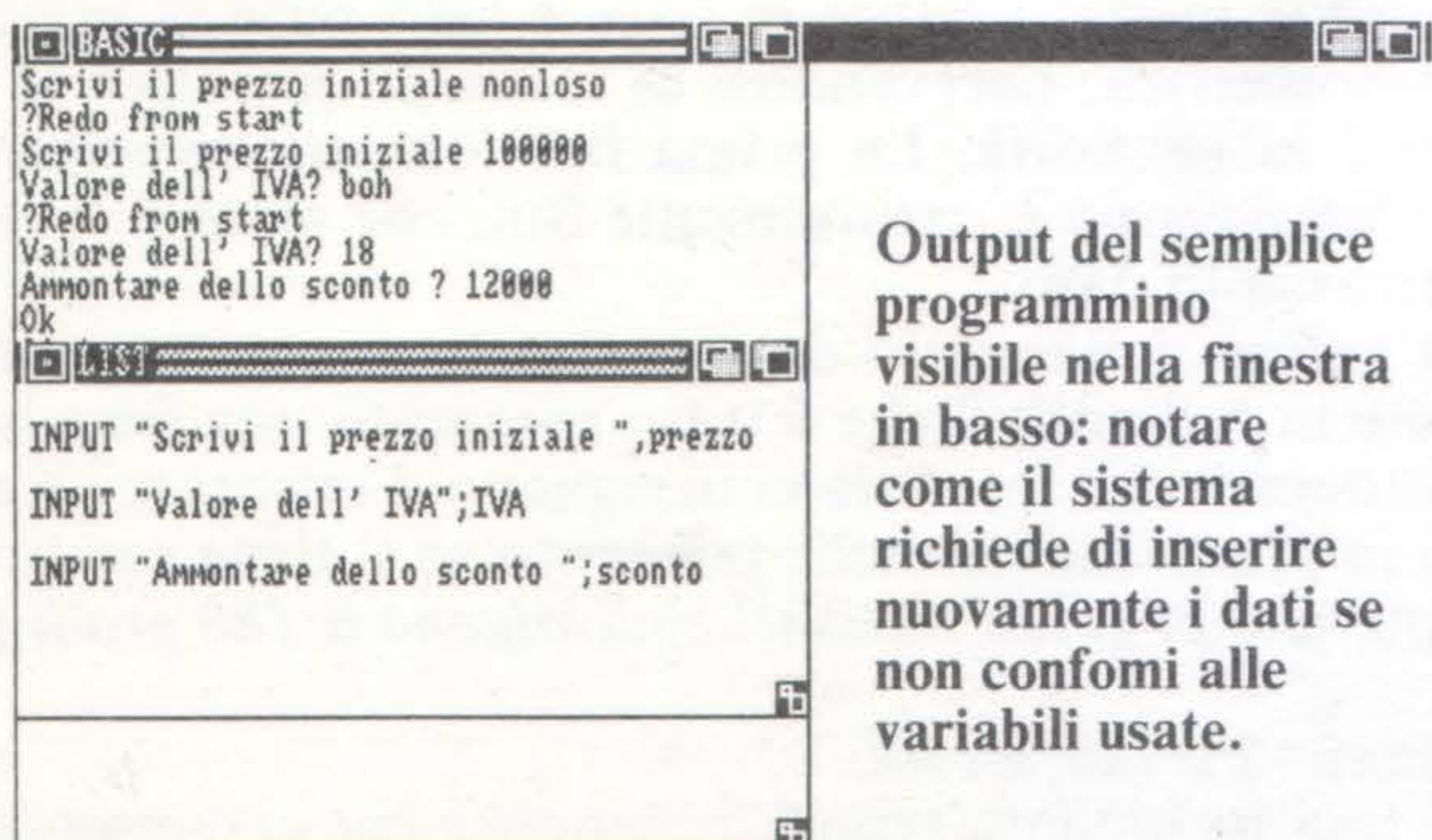
Infine, c'è l'elevamento a potenza, rappresentato dal simbolo «^» (corrispondente all'accento circonflesso o ad una piccola lambda o ad una v rovesciata).

E le radici? Già, le radici non ci sono, ma l'albero sta in piedi lo stesso; si consideri infatti che:
n^(1/m)
(enne elevato al reciproco di emme)
equivale ad estrarre la radice *emmesima* di n; il problema è risolto.

Per la verità esiste la radice quadrata, ma essa non viene considerata un operatore aritmetico, bensì una funzione (vedi oltre), una scelta dei creatori del Basic alla quale dobbiamo sottostare.

Come nell'algebra classica, esistono delle priorità di esecuzione.

Prima gli elevamenti a potenza
Poi moltiplicazioni e divisioni
Quindi somme e sottrazioni.

A parità di *ordine gerarchico* (modo pomposo per dire priorità) le espressioni vengono valutate da sinistra verso destra, ma di questo non dobbiamo preoccuparci. Ancora come nell'algebra classica, le parentesi possono essere utilizzate per far sì che l'esecuzione segua un ordine diverso da quello specificato dalla gerarchia algebrica.

Output del semplice programmino visibile nella finestra in basso: notare come il sistema richiede di inserire nuovamente i dati se non confomi alle variabili usate.

A differenza invece dell'algebra classica, non si usano parentesi quadre o grafe, solo tonde.

I livelli più interni vengono ovviamente valutati prima di quelli esterni. Per ogni parentesi che apriamo ne deve essere chiusa una, prima o poi, pena la comparsa di messaggi d'errore.

Se mettiamo una parentesi inutile (in un punto cioè in cui l'ordine di valutazione sarebbe comunque quello desiderato) non si hanno problemi, è come se non fosse stata messa; è tuttavia opportuno lesinare le parentesi per non rendere l'espressione confusa da interpretare (per l'operatore, non per il computer). Facciamo qualche esempio.

```
Let x1=(—b+(b^2—4*a*c)^(1/2))/(2*a)
Let x2=(—b—(b^2—4*a*c)^(1/2))/(2*a)
```

è la risoluzione delle equazioni di secondo grado, l'avevate riconosciuta, vero?

Notate l'uso delle parentesi per forzare l'ordine di esecuzione delle operazioni; notate anche che, in molti casi, la gerarchia algebrica rende superflue eventuali altre parentesi. Facciamo qualche altro esempio:

```
Let Circ=Raggio*2*3.14
Let Area=Raggio^2*3.14
Print"Il Volume è uguale a";n*R*T/P;" Litri"
```

Il punto e virgola separa i diversi elementi di un'istruzione «Print» e gli elementi non racchiusi fra virgolette vengono valutati (cioè calcolati) e ne viene stampato il risultato.

Un piccolo trucco di programmazione: poiché l'*algoritmo* (cioè la serie di calcoli, eseguiti a livello del linguaggio macchina) dell'elevamento a potenza è sensibilmente più complesso e, di conseguenza, più lento di quello della moltiplicazione, quando occorre elevare un numero al quadrato o al cubo è conveniente moltiplicarlo per se stesso due o tre volte anziché utilizzare l'operatore elevamento a potenza, cioè:
anziché x^2 è meglio x*x
ed anziché x^3 è preferibile x*x*x

Il risultato non cambia, ma l'esecuzione è sensibilmente più veloce, soprattutto se questa operazione avrà la ventura di dover essere ripetuta diverse volte all'interno del programma.

SPERIAMO CHE FUNZIONI

Avete probabilmente già capito che l'eterno triangolo Comandi-Istruzioni-Variabili cui si accennava la volta scorsa non rappresenta l'intera gamma di elementi che si possono trovare in Basic. Dopo gli operatori aritmetici, abbiamo infatti anche le Funzioni. Chi ha studiato a scuola, ad esempio, le funzioni trigonometriche, può farsi

un'idea di quello di cui stiamo parlando. Per gli altri diremo che le funzioni sono un po' come gli elettrodomestici che si usano in cucina, il frullatore, il tritacarne o il gelataio: da una parte si mette la materia prima e dall'altra esce il prodotto lavorato, la cui forma dipende dalla macchina che è stata usata e la cui sostanza dipende dal materiale utilizzato.

Così le funzioni; esse ci permettono di passare da un *argomento* (la materia prima delle funzioni) ad un *risultato* (il prodotto lavorato). Anche qui il risultato dipenderà sia dalla funzione utilizzata che dall'argomento.

Breve programmino di assegnamento con operatori aritmetici: l'unico particolare degno di rilievo è la radice ennesima ottenuta come elevamento a potenza.

In Basic esistono svariati tipi di funzioni che ricoprono l'intero arco di operazioni possibili ed immaginabili e, anche se si riesce ad ipotizzare una funzione non prevista, è sempre possibile crearcela su misura con un'istruzione *ad hoc*.

Come per gli elettrodomestici, l'argomento deve essere coerente con il tipo di funzione utilizzata; tornando all'esempio casalingo, se mettiamo carne di manzo nel gelataio il risultato non sarà probabilmente conforme alle nostre aspettative. (Se inventate il gelataio di manzo fatecelo sapere, provvederemo alla rettifica).

Il risultato di una funzione può essere a tutti gli effetti equiparato a variabili o costanti dello stesso tipo.

Per cominciare, non essendo esse stesse istruzioni, hanno bisogno di un'istruzione in cui comparire; il risultato può quindi essere scritto sullo schermo (Print), può essere usato in espressioni, può essere assegnato come

valore a variabili (Let), può essere persino usato come argomento di altre funzioni.

Per contro, l'argomento di una funzione può essere

Semplici funzioni matematiche ottenute con assegnamento a variabili, oppure stampate direttamente sul video con l'istruzione Print.

una costante, una variabile, un'espressione contenente costanti e/o variabili o, come si è detto, il risultato di un'altra funzione.

In genere l'argomento di una funzione viene posto fra parentesi a seguire la funzione stessa; quando si tratti di un argomento stringa, viene posto anche fra virgolette.

Facciamo ora la conoscenza di alcune delle funzioni più importanti del Basic; non le descriveremo tutte in questa sede, riservandoci di trattare le altre al momento opportuno.

LE FUNZIONI MATEMATICHE

Sono le funzioni più classiche, accettano argomenti numerici e sfornano risultati numerici. Ecco che qui troviamo la nostra radice quadrata, rappresentata dalla funzione SQR (da SQuare Root, che significa appunto «Radice quadrata»):

Print Sqr (64)

Scriverà sul video il numero 8,

Let rad=Sqr(10*4+10—1)

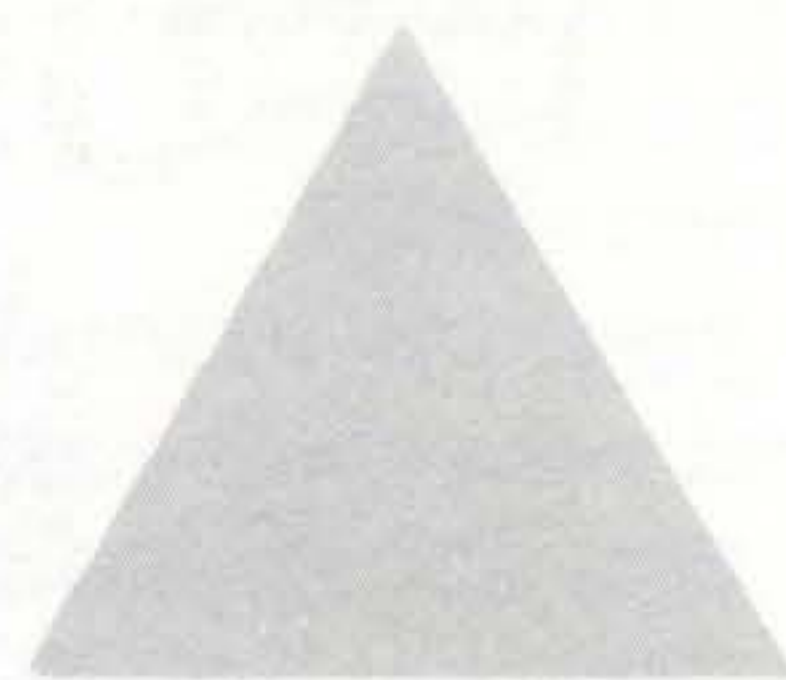
Assegnerà alla variabile Rad il valore 7,

Let Ipotenusa=Sqr (Cateto1^2+Cateto2^2)

vi lasciamo immaginare quale teorema rappresenti.

Si accennava alle funzioni trigonometriche, disponibili infatti in Basic.

Chi non sa cosa sia un seno (in trigonometria, ovvia-

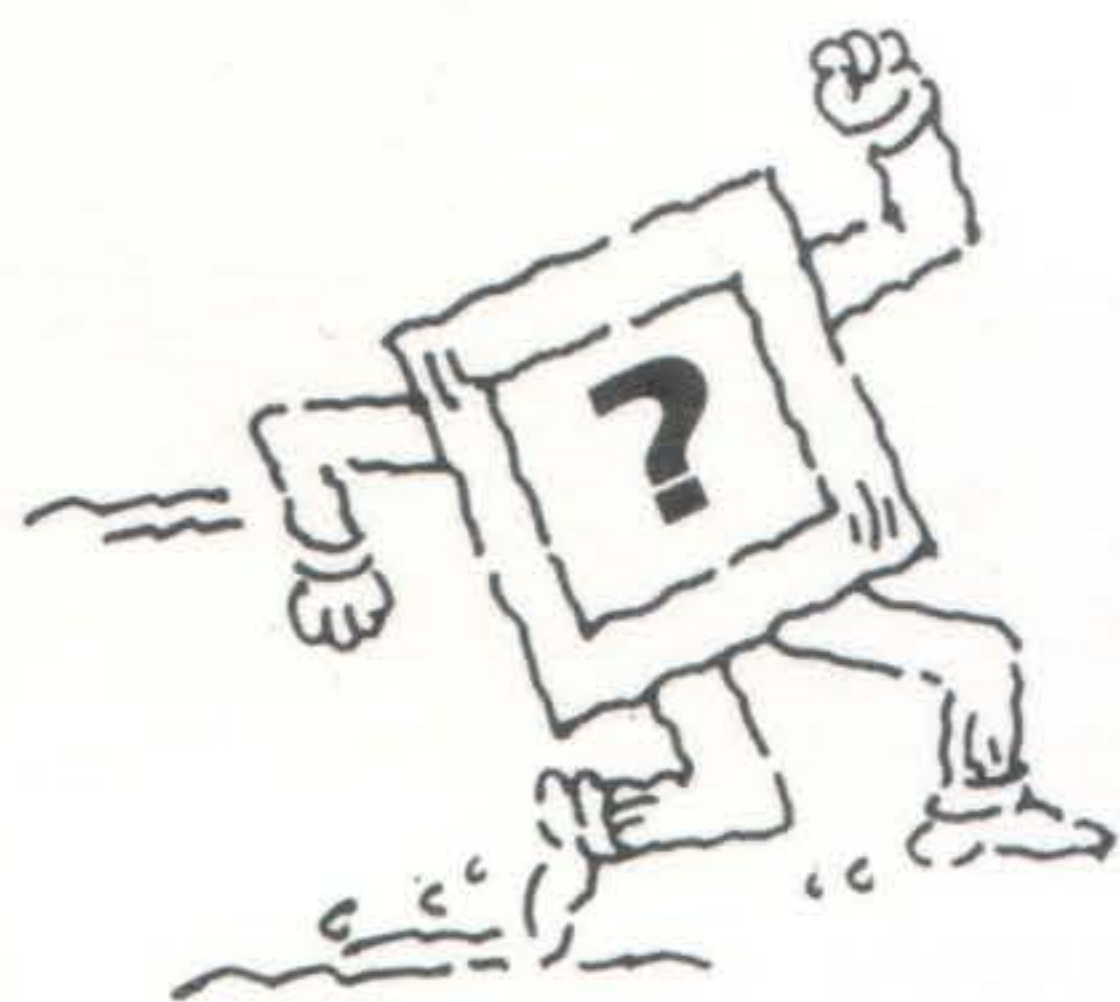
mente) si metta il cuore in pace e salti pure la sezione trigonometrica, riservandosi di tornarci quando e se ne sentirà la necessità. La prima funzione trigonome.rica che descriviamo è naturalmente Sin, che calcola il seno di un angolo dato.

Il valore dell'angolo deve essere espresso necessariamente in radianti, il che è tutto sommato una fregatura considerato che a scuola ci insegnano a misurare gli angoli in gradi, ma neanche poi tanto se si tiene sempre a mente che *pi greco radianti equivalgono a 180 gradi* per cui:

radianti=Pi/180*Gradi

È una formuletta semplice semplice che ci consente di passare dai gradi ai radianti; ovviamente «Pi» rappresenta il pi greco che vale 3,14..... (potete metterlo lungo a piacere a seconda della precisione che vi interessa).

E, sempre a proposito della precisione, il risultato sarà dello stesso tipo dell'argomento, in singola precisione se esso era in singola, in doppia se era in doppia e, trattandosi di funzioni trigonometriche, in questo caso sarà sempre reale, ovvero con la virgola. Facciamo alcuni esempi:

Print "Il seno di Pi greco mezzi è ";Sin (3.14/2)
Lato=Ipotenusa*Sin(alfa)

Fratello gemello del seno è il coseno, espresso dalla funzione Cos; vale per esso tutto ciò che si è detto per Sin. Abbiamo infine la tangente (non nel senso che alla

fine ci han dato la bustarella) espressa dalla funzione Tan, per la quale pure vale ciò che è stato detto per il seno.

Il sistema mette anche a disposizione l'arcotangente con la funzione ATN: l'argomento deve essere in radianti ed il risultato è anch'esso in radianti, compreso fra — Pi/2 e Pi/2. Per il resto la musica non cambia; a noi non è mai capitato di usarlo, ma lo descriviamo per completezza.

Chi fosse abbastanza... pervertito da sentire il bisogno delle funzioni inverse, iperboliche e chi più ne ha più ne metta, troverà le espressioni adatte per calcolarle nell'appendice E del manuale AmigaBasic.

Avete notato probabilmente che abbiamo usato, per le funzioni come per le istruzioni, caratteri minuscoli-maiuscoli o misti, a seconda di come ci andava in quel momento.

VEDIAMO QUALCHE ESERCIZIO

Tale promisquità è voluta: anche le funzioni vengono riconosciute indipendentemente dal fatto che vengano scritte maiuscole o minuscole; il sistema però, non appena riconosce una funzione, la riscrive in caratteri maiuscoli.

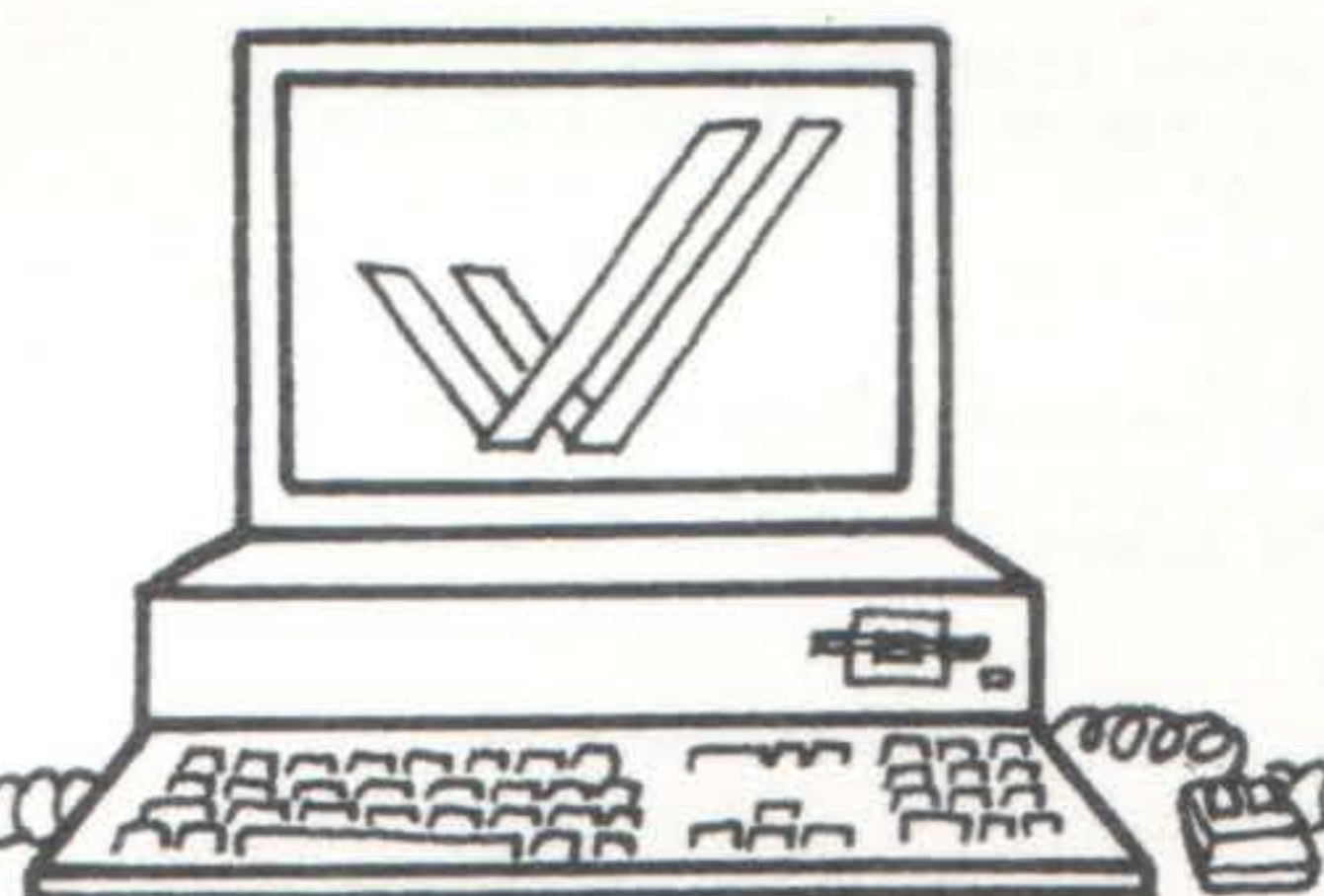
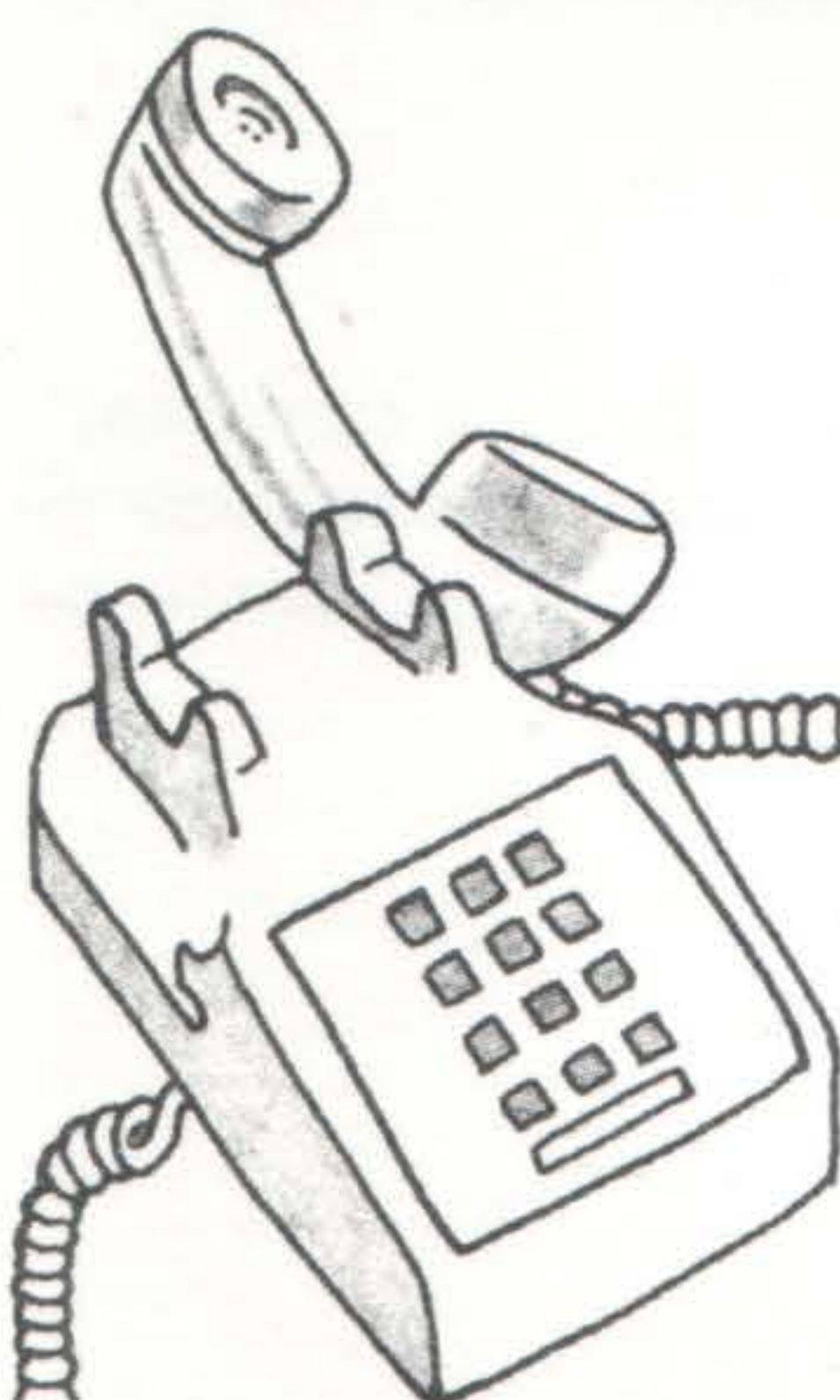
Se una funzione o un'istruzione non vengono riscritte subito (cioè alla pressione del tasto «RETURN») in maiuscolo dal sistema, vuol dire che c'è qualcosa che non va: controllatele e correggetele se necessario, altrimenti verrà generato un messaggio d'errore al momento dell'esecuzione.

Torneremo sulle funzioni nella prossima puntata. Per concludere questa, vi proponiamo alcuni esercizi:

- scrivete un programma che chieda di inserire un numero e ne calcoli quindi il quadrato, il cubo, la radice quadrata, la radice cubica, il seno, il coseno e la tangente, e che scriva il risultato accompagnato da un messaggio del tipo:

Il quadrato di ... è ...

- scrivete un programma che chieda il valore dei cateti di un triangolo rettangolo e calcoli e scriva l'ipotenusa.

Per le soluzioni, arrivederci sul prossimo fascicolo. ■

"C"

I PUNTATORI

3ª puntata

di ALESSANDRO BOSSI

Oggi parliamo di puntatori e dell'enunciato IF. Abbiamo visto nella scorsa puntata i tipi di dati, fra cui un misterioso tipo «pointer (*)», che occupa 32 bit di spazio. Vediamo meglio cos'è. I dati occupano uno spazio fisico nella memoria del calcolatore; durante l'esecuzione di un programma potrebbe essere utile, o addirittura indispensabile, sapere dove sono stati memorizzati fisicamente certi dati, sapere cioè il loro indirizzo in memoria. I puntatori sono stati creati apposta per questo. Abbiamo già utilizzato l'operatore di indirizzo

```
Amiga_Byte 3:Corso_C
2> type prog.4
void main()
{
int numero;

printf("Inserisci un numero: ");
scanf("%d",&numero);
if (numero == 0)
   {
   printf("Il numero inserito e' 0\n");
   printf("Quindi eseguo solo il corpo di IF\n");
   }
else
   {
   if (numero < 0)
      printf("Il numero e' minore di zero\n");
   else
      printf("Il numero e' maggiore di zero\n");
   }
printf("Fine\n");

}
2>
```

Stampa del file Prog. 4, contenuto nel dischetto allegato a questo stesso fascicolo, del programmino che controlla se un numero è maggiore o minore di 0.

(&) nei nostri esempi, tutte le volte che abbiamo adoperato la funzione scanf(). Riprendiamo allora l'ultimo esempio fatto, che si trova sul disco allegato a questo fascicolo nella directory «corso_c» sotto il nome «prog.1».

```
void main()
{
int n;
char prova[10];

printf("Inserisci una stringa di 9 caratteri: ");
scanf("%s", &prova);
printf("\nHai inserito: %s\n",prova);
for (n=0; n < 10; n++)
   printf("elemento %d: %c, ASCII: %d\n",n,prova[n],
                          prova[n]);
}
```

All'inizio del programma abbiamo dichiarato un array di 10 caratteri chiamato «prova» poi, nel chiamare la funzione scanf(), abbiamo usato come parametro «&prova». L'operatore di indirizzo & serve per trovare l'indirizzo della variabile a cui si prepone. In questo caso, non abbiamo passato a scanf() l'array di caratteri, bensì il puntatore a questo array, cioè l'indirizzo in memoria del primo elemento. Durante l'esecuzione di un programma viene riservato (allocato) dello spazio in memoria per le variabili utilizzate dal programma. Per ogni variabile viene allocata una quantità di memoria sufficiente a contenere il massimo valore raggiungibile da quel tipo di variabile. Una volta riservata, questa zona di memoria non cambia. Rimane disponibile durante l'esecuzione di tutto il programma (o di parti del programma; in seguito specificheremo meglio). La scrittura «&prova» rappresenta allora una costante di puntatore, perché durante l'esecuzione non cambia. La variabile di tipo puntatore si dichiara utilizzando l'asterisco, che in questo caso non va confuso con l'operatore di moltiplicazione.

```
char *inp;
```

Questa dichiarazione dice: 'inp è un puntatore a tipo char'. Significa che «inp» punta a valori di tipo char. Potremmo riscrivere il nostro programma (sul nostro dischetto, "corso_c/prog.2") in questo modo:

```
void main()
{
int n;
char prova[10];
char *inp;

inp = &prova;
printf("Inserisci una stringa di 9 caratteri: ");
scanf("%s", inp);
printf("\nHai inserito: %s\n",inp);

for (n=0; n < 10; n++)
   {
   printf("elemento %d: %c, ASCII: %d\n",n,*inp,*inp);
   inp++;
   }
}
```

Abbiamo dichiarato un array di 10 caratteri (char prova[10]) ed un puntatore a tipo carattere (char *inp). L'istruzione «inp = & prova» assegna alla variabile «inp» il puntatore alla variabile «prova». Abbiamo detto che «&prova» rappresenta una costante di puntatore, quindi è lecito assegnare il valore di questa costante alla variabile «inp», che adesso contiene il puntatore al primo elemento di «prova». A questo punto, «&prova» e «inp» puntano entrambi all'array «prova», quindi pos-

Tabella A

Variabile	Locazione	Valore	Dimensione
caramelle	$5a3430	124	32 bit
puntacar	$674200	$5a3430	32 bit

Tabella A. Rappresentazione in memoria delle variabili «caramelle» e «puntacar».

siamo tranquillamente utilizzare la variabile «inp» come argomento da passare a scanf() e printf(). Nella chiamata alla funzione printf() contenuta nel ciclo for, torna l'asterisco. Questa volta non è una dichiarazione di puntatore, né una moltiplicazione: prende il nome di operatore di indirizzamento indiretto, e serve a trovare il valore di una variabile di cui conosciamo il puntatore. Supponiamo di avere una variabile, di tipo int, che rappresenti il numero di caramelle di un barattolo:

```
int caramelle;
caramelle = 124;
```

Aggiungiamo la dichiarazione di una variabile di tipo puntatore ad int:

```
int caramelle, *puntacar;
```

Adesso assegnamo a «puntacar» il valore del puntatore alla variabile caramelle:

```
puntacar = &caramelle;
```

QUANTE CARAMELLE IN UN BARATTOLO

Supponiamo che in memoria la variabile caramelle, che occupa 32 bit (4 bytes), si trovi alla locazione $5a430 (con il suffisso $ si indica un numero in notazione esadecimale e, salvo avviso contrario, useremo sempre questa notazione per gli indirizzi). Allora «puntacar» contiene $5a3430. Adesso vogliamo sapere quante caramelle sono contenute in un barattolo, ma non ci ricordiamo più della variabile «caramelle». Abbiamo però il puntatore a questo numero che vogliamo leggere, allora usiamo l'operatore di indirizzamento indiretto e scriviamo «*puntacar».
printf("caramelle: %d",*puntacar); Questo ci restituisce 124, il numero dal quale siamo partiti. La rappresentazione in memoria delle variabili «caramelle» e «puntacar», si trova nella tabella A. Un'osservazione sull'ultima istruzione del programma ci

Tabella B

Operatore	Significato
==	uguale
!=	diverso
<=	minore o uguale
>=	maggiore o uguale
<	minore
>	maggiore

Tabella B. Elenco degli operatori relazionali del C.

porta a riflettere: «inp++» significa 'incrementa inp'; quando il compilatore trova questa istruzione fà una serie di valutazioni: «inp» è un puntatore, allora devo incrementarlo affinché punti all'oggetto successivo. Ma a quale oggetto punta? Punta al tipo char, che è composto da un byte, quindi devo incrementare «inp» di un byte. Se «inp» fosse stato un puntatore a qualche altro tipo di dati, l'incremento (o il decremento) sarebbe stato relativo al tipo di dati in questione. Ad esempio, dichiarando:

```
int *conta;
```

l'istruzione:

```
conta++
```

incrementa il valore di «conta» di 4 alla volta.

Apriamo una parentesi sull'enunciato condizionale IF.

Questa istruzione ci permette di eseguire un gruppo di istruzioni se, e solo se, sono verificate determinate condizioni. Vediamo subito un esempio pratico (sul nostro dischetto, "corso_c/prog.3"):

```
void main()
{
char risposta;

printf("Vuoi continuare ? (S/N) ");
scanf("%c",&risposta);
if (risposta == 'S')
    printf("\nOk, continuiamo!\n");
else
    printf("\nNon vuoi continuare: mi fermo. \n");
printf("Il programma è finito\n");

}
```

Il programma attende un carattere dalla tastiera, e lo memorizza nella variabile char. L'istruzione IF valuta l'espressione contenuta nella parentesi e, se questa è vera, esegue la prima istruzione printf(), altrimenti (istruzione ELSE) esegue la seconda printf(). Dopo la seconda printf() l'enunciato condizionale termina, ed in tutti e due i casi viene eseguito il resto del programma, ovvero la terza istruzione printf(). Proviamo a lanciare il programma, e rispondiamo battendo una 'S' maiuscola. Il programma stampa «OK, continuiamo!». Se rispondiamo con un'altra lettera qualsiasi, compresa la 's' minuscola, il programma stampa «Non vuoi continuare: mi fermo.». In entrambi i casi il programma segnala la sua fine stampando «Il programma è finito». Nel corpo dell'enunciato IF possiamo mettere una sola istruzione, come abbiamo fatto nel nostro programma, oppure più istruzioni, raggruppandole nella solita coppia di parentesi graffe. L'istruzione ELSE è facoltativa e l'istruzione (o il corpo di istruzioni) che la segue viene eseguita solo quando l'espressione condizionale non è soddisfatta. Vediamo adesso gli operatori relazionali, che sono elencati nella tabella B. Il primo operatore «==» lo abbiamo usato nel no-

```
Amiga_Byte 3:Corso_C
2> dir
  .fastdir                        .info
  prog.1                          prog.1.info
  prog.2                          prog.2.info
  prog.3                          prog.3.info
  prog.4                          prog.4.info
  prog.5                          prog.5.info
2>
2>
2> type prog.1
void main()
{
int n;
char prova[10];

printf("Inserisci una stringa di 9 caratteri: ");
scanf("%s", &prova);
printf("\nHai inserito: %s\n",prova);

for (n=0; n < 10; n++)
   printf("elemento %d: %c, ASCII: %d\n",n,prova[n],prova[n]);
}
2>
```

Elenco della directory Corso_c che trovate nel nostro dischetto: ogni singolo programmino ha il suo file .info che gli permette di essere visualizzato anche da Workbench.

stro programma. La variabile «risposta» è uguale al carattere 'S'? Sì, esegui il corpo di IF; No, esegui il corpo di else. Attenzione a non confondere l'operatore di assegnamento «=», che assegna il valore di destra alla variabile di sinistra, con l'operatore relazionale di uguaglianza «==» che compara fra loro i due operandi. Quando viene elaborata una espressione relazionale fra due operandi, il risultato è FALSO (0) o VERO (1). Quando l'istruzione IF va a guardare il risultato dell'operazione, considera FALSO il valore 0, mentre considera VERO non solo il valore 1, ma qualsiasi altro valore diverso da zero. Se allora, confondendo «=» con «==», scriviamo:

```
if (risultato = 'S')
```

l'espressione verrà sempre considerata VERA, in quanto assegnamo alla variabile «risultato» il valore ASCII del carattere «S», che è diverso da zero. È perfettamente «legale», invece, sostituire all'espressione condizionale una variabile:

```
if (risultato)
```

In questo caso, la 'condizione' è sempre verificata (VERO), per qualsiasi valore di «risultato» che sia diverso da zero.

```
Tabella C

Operatore   Significato

   &&       AND (e)
   ||       OR  (oppure)
   !        NOT (negazione)
```

Tabella C. Elenco degli operatori logici del C.

Complichiamo un po' l'ultimo programma (sul nostro dischetto, "corso_c/prog.4"):

```
void main()
{
int numero;

printf("Inserisci un numero: ");
scanf("%d",&numero);
if (numero == 0)
  {
  printf("Il numero inserito è 0\n");
  printf("Quindi eseguo solo il corpo di IF\n");
  }
else
  {
  if (numero < 0)
    printf("Il numero è minore di zero\n");
  else
    printf("Il numero è maggiore di zero\n");
  }
printf("Fine\n");

}
```

In questo caso, il corpo di IF viene eseguito solo se inserite il numero 0, altrimenti viene eseguito il corpo di ELSE, nel quale si trova un'ulteriore istruzione IF abbinata alla sua ELSE. Schematicamente:

```
if (prima espressione)
  primo enunciato
else
  if (seconda espressione)
  secondo enunciato
  else
  terzo enunciato
```

Quindi, se la prima espressione (numero == 0) è vera, allora viene eseguito il primo enunciato. Se la prima espressione è falsa, ma la seconda è vera, viene eseguito il secondo enunciato. Se entrambe le espressioni sono false, viene eseguito il terzo enunciato. Torniamo adesso al programma che ci chiedeva "Vuoi continuare ?" ("prog.3"); questo accettava come unica risposta il carattere «S» maiuscolo. Modifichiamolo in modo che accetti come risposta affermativa anche la «s» minuscola (sul nostro dischetto, "corso_c/prog.5"):

```
void main()
{
char risposta;

printf("Vuoi continuare ? (S/N) ");
scanf("%c",&risposta);
if (risposta == 'S' || risposta == 's')
  printf("\nOk, continuiamo!\n");
else

  printf("\nNon vuoi continuare: mi fermo.\n");
printf("Il programma è finito\n");
}
```

Adesso accetterà entrambe le esse. Abbiamo introdotto l'operatore logico "||" (due barrette verticali, ottenute premendo due volte il tasto della singola barretta verticale, situato in posizioni differenti, a seconda della tastiera che state utilizzando. Generalmente si ottiene premendo SHIFT ed il tasto "\" backslash). Gli operatori logici del C sono elencati nella tabella C. Questi operatori permettono di combinare insieme più espressioni relazionali. Nel nostro caso la traduzione in linguaggio corrente della nostra istruzione IF suona così:

se "risposta" è uguale a "S", OPPURE, se "risposta" è uguale a "s", allora esegui l'enunciato che segue.

Se avessimo usato l'operatore "&&", la condizione non sarebbe mai stata soddisfatta, ed il programma avrebbe sempre seguito il ramo ELSE. È chiaro infatti che l'espressione condizionale:

se "risposta" è uguale a "S", E CONTEMPORANEAMENTE, se "risposta" è uguale a "s", allora...

non sarà mai VERA, perché è impossibile che una variabile sia contemporaneamente uguale a due valori differenti fra di loro. In pratica la tabella della verità dice che:
(espressione1 && espressione2 && ...)
è VERA solo se tutte le espressioni sono vere.
(espressione1 || espressione2 || ...)
è VERA solo se almeno una delle espressioni è vera.
(!espressione1)
è VERA se espressione1 è falsa.

Non abbiamo usato le parentesi per dividere le diverse espressioni in quanto superflue, perché gli operatori logici hanno una priorità più bassa di quelli relazionali. Tuttavia non guasta, in alcuni casi, per migliorare la leggibilità di un programma, fare uso di parentesi superflue per raggruppare e separare visivamente alcuni gruppi di espressioni.

■

Manipoliamo i file

Impariamo ad approfittare della potenza del cambio di directory e ad usare tutte le opzioni del comando type per visualizzare correttamente ogni tipo di file, sia di testo che d'altro.

di RICKY SWORD

Abbiamo già detto che per raggiungere un qualsivoglia file o una directory annidata all'interno di altre directory fino a livelli anche piuttosto profondi, è necessario sempre specificare il *percorso* (in inglese «path») che il sistema deve compiere per raggiungerli. La cosa può, alla lunga, diventare noiosa.

Dovendo ad esempio agire sui diversi file contenuti nel cassetto BasicDemos del disco Extras (ad esempio leggerli, copiarli, cancellarli etc...) occorrerà, per ogni operazione, che si specifichi:

Comando Extras:BasicDemos/Nome_file

e che si ripeta il percorso per ogni file e per ogni comando.

È evidente che questo procedimento, oltre che essere noioso e poco ergonomico, ci causa un discreto dispendio di tempo.

Meglio sarebbe piazzarci una volta per tutte nella directory desiderata (in questo caso BasicDemos) istruendo il sistema che, da quel momento in poi e a meno di specificare altrimenti, quella sarà la nostra directory corrente.

Il metodo per ottenere ciò naturalmente esiste, ed è rappresentato dal comando **Cd** (abbreviazione di Change Directory, ovvero «Cambia Directory»).

Per piazzarci nella directory desiderata basta far seguire al comando citato il path, considerato dalla directory principale, ovvero quella del nostro Workbench.

Così:

Cd Extras:BasicDemos

ci porterà nella directory BasicDemos del disco extras.

```
Lavoro:amiga_3
2>
2> cd df1:amiga_3
2> cd df1:amiga_3/recensioni
2> cd sys:
2> cd df1:
2> cd sys:devs/printers
2> cd lavoro:
2> cd lavoro:amiga_3/recensioni
2> cd /
2> cd df0:devs
2> cd df1:amiga_3
2>
```

Esempi vari sull'uso del comando CD per passare da una directory ad un'altra. Se si indicherà SYS: come dispositivo, il dos ci porterà nella root directory del disco di sistema.

Adesso, con un comando come *Dir* o *List*, non seguito da alcun parametro, otterremo l'elenco dei file della directory in cui ci troviamo e non più quello della root directory.

A Cd possiamo far seguire alcuni caratteri addizionali; per cominciare, Cd non seguito da alcun carattere scrive sul video il percorso compiuto. Il carattere «/» (barra di diviso) porta il sistema al livello precedente, cioè alla directory che contiene quella in cui ci troviamo. Ad esempio:

Cd df0:devs/printers

Ci porta nella directory printers della directory devs del disco nel drive 0 e

Cd /

Ci riporta nella directory devs.

Altro carattere consentito è «:» (due punti) che porta il sistema alla root directory del disco selezionato così:

Uso dell'opzione N del comando Type per aggiungere automaticamente il numero di linea al testo in output.

Semplice Type di un file di testo, un classico Readme preso dal disco Extras. Notare la presenza della barra in fondo alla finestra per bloccare lo scroll.

Cd df1:BasicDemos

Ci porta nella directory BasicDemos del drive 1 e

Cd :

Ci porta nella root directory del medesimo drive.

Tenete presente che un percorso specificato nell'ambito di un comando ha la priorità rispetto alla posizione in cui ci ha messo il comando Cd; così, anche qualora ci si trovi nella directory BasicDemos del disco Extras, il comando:

List df0:devs

ci darà la lista della directory Devs del disco nel drive 0.

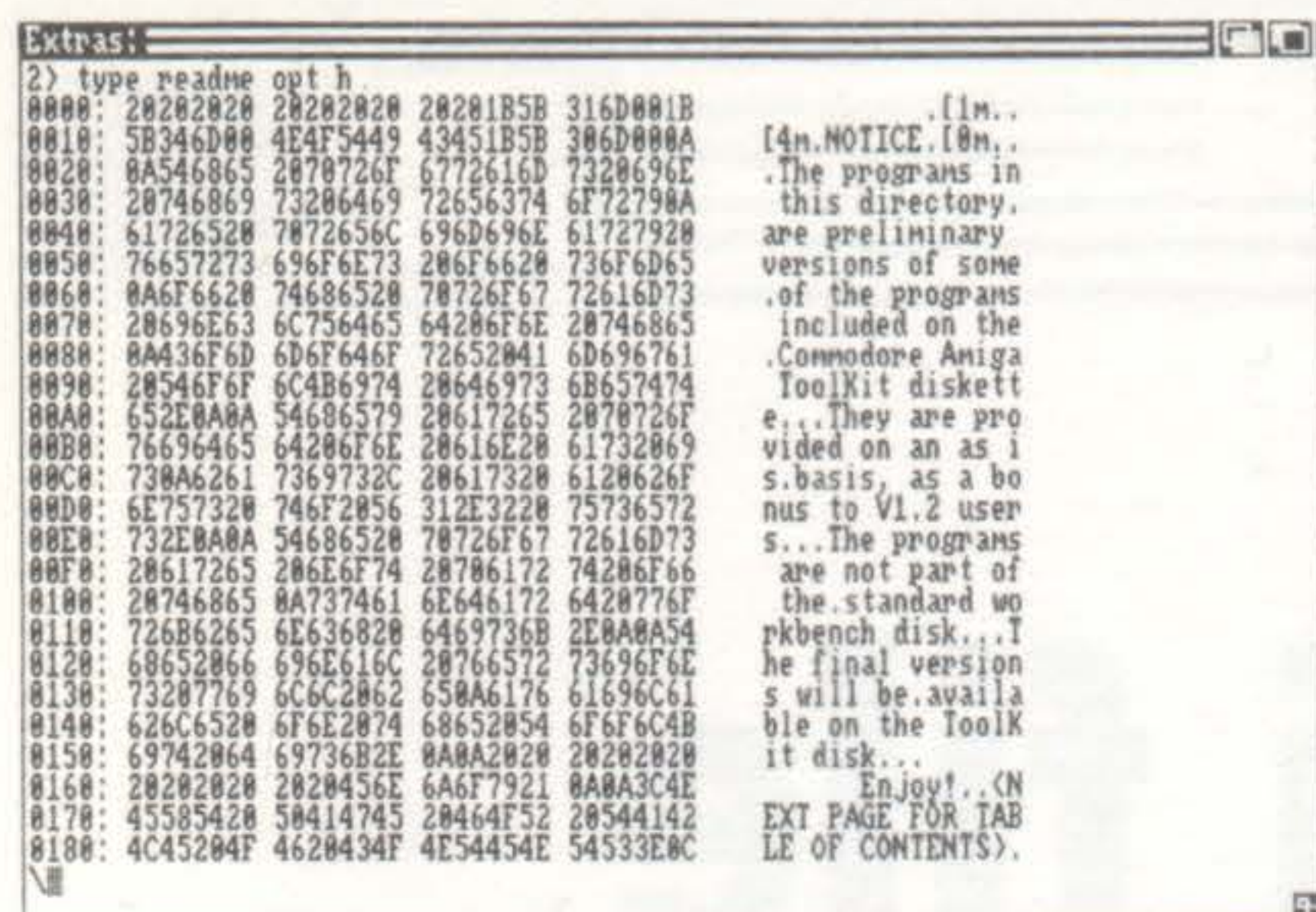
Inoltre il comando:

Cd Sys:

ci porterà sempre nella root directory del disco di sistema, ovvero di quel disco che è stato inserito subito dopo il Kickstart o, all'accensione, nell'Amiga 500 e 2000.

MA GUARDA CHE *TYPE*

Fino ad ora abbiamo esaminato comandi relativi alle directory: Dir e List per elencare i files in essi contenuti e Cd per raggiungere qualcuna in particolare.

Si accennava però alle operazioni che

Per esaminare un qualsiasi file, non necessariamente di testo, è consigliabile servirsi dell'opzione H, sempre dopo il comando Type naturalmente.

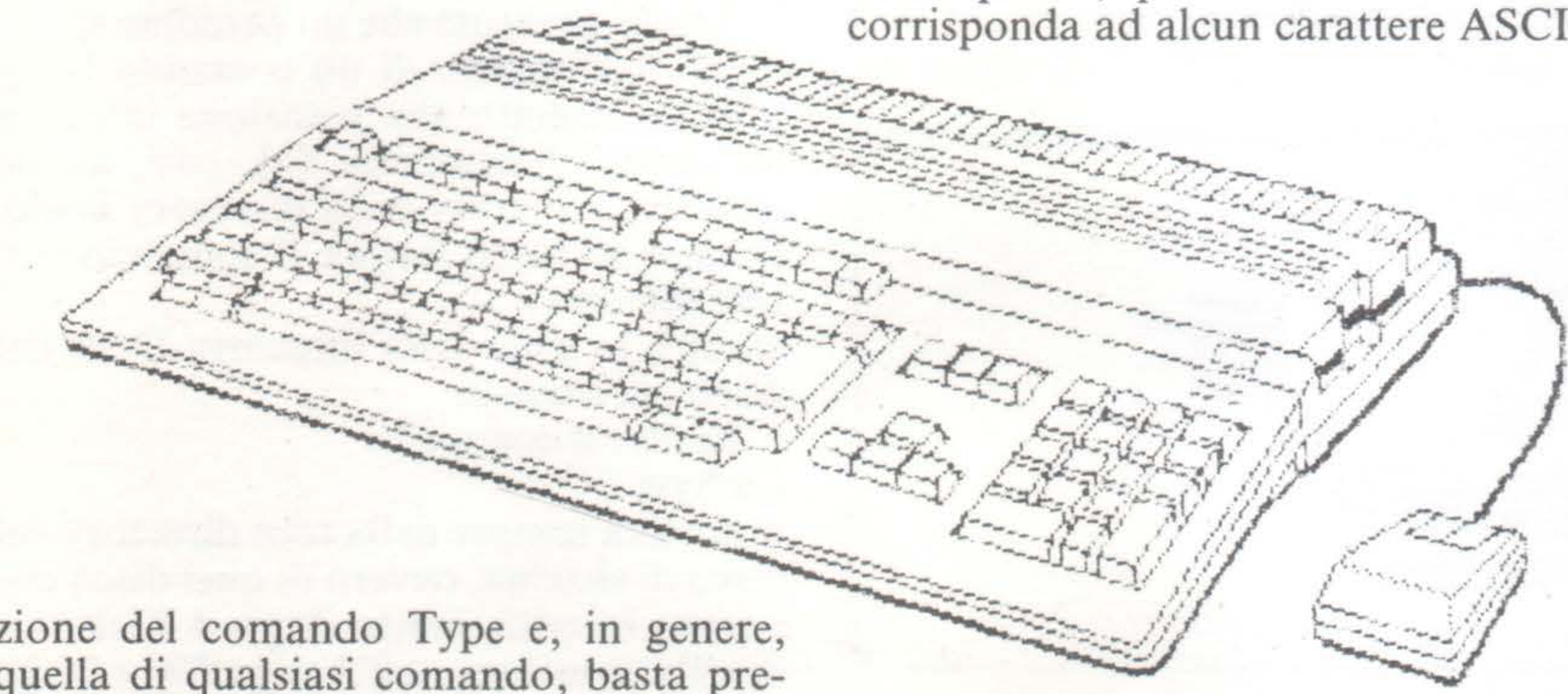
si possono fare sui files.

I files di testo, quelli cioè che contengono esclusivamente testo, possono essere letti direttamente da Cli con il comando **Type** (che significa «Stampa» e viene riferito, in inglese comune, più che altro alle macchine da scrivere).

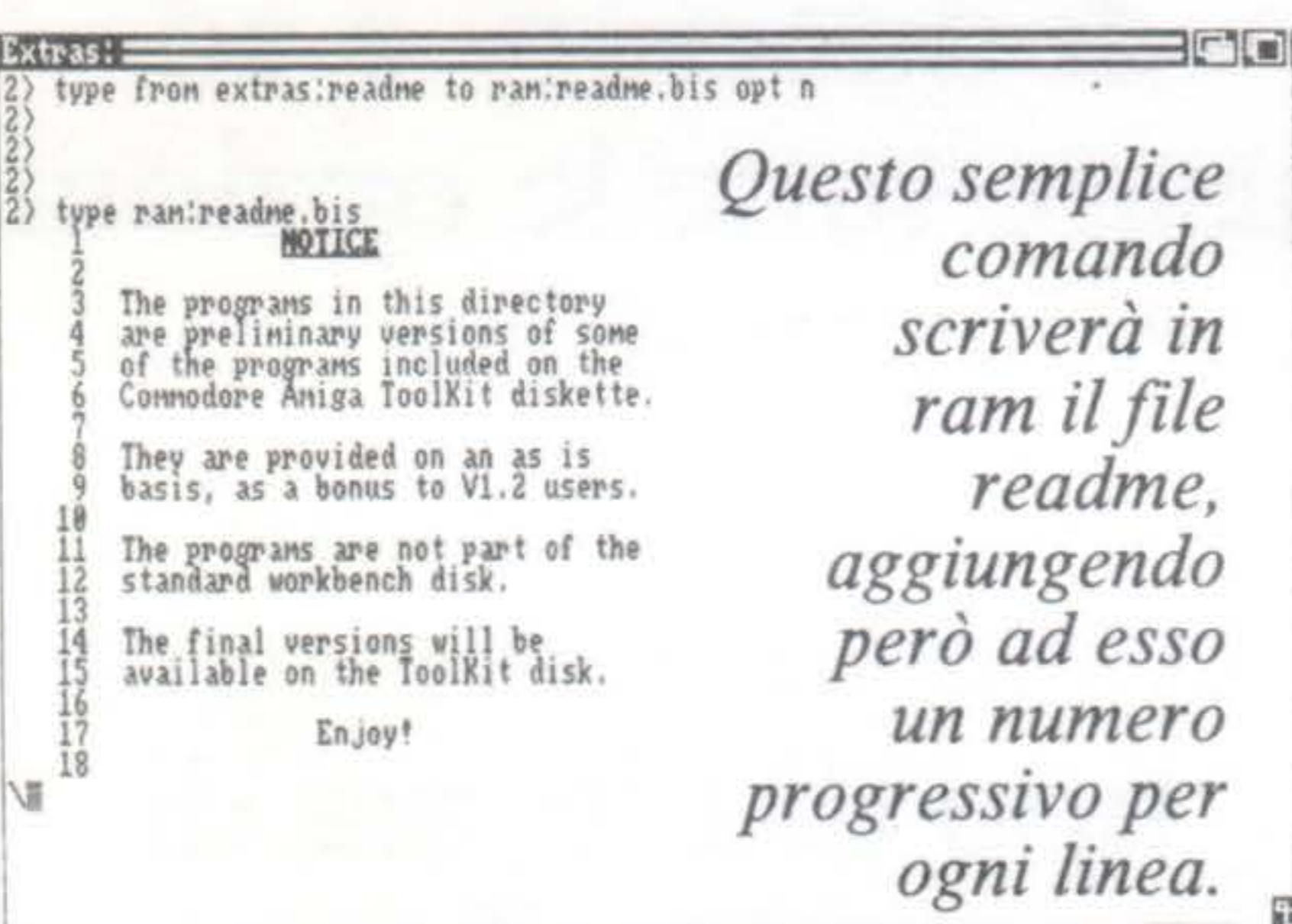
Ad esempio, trovando in un disco un file testo di nome «ReadMe» o «Readme.doc» (succede molto spesso) il comando:
Type ReadMe
(o Type Readme.doc, a seconda del caso) scriverà sul video l'intero contenuto del file di testo, che scivolerà di una linea verso l'alto con un sistema detto di *scrolling* non appena l'ultima linea dello schermo viene completata.

Poiché lo scrolling è in genere molto veloce, per sospenderlo momentaneamente basta premere un tasto qualsiasi (di solito la barra spaziatrice). Scrivendo qualcosa da tastiera infatti si interrompe sempre l'esecuzione di qualunque comando, non solo type, che abbia un output su video; per riprendere l'esecuzione basta cancellare (Back Space) ciò che si è testé inserito.

Per interrompere definitivamente l'a-

zione del comando Type e, in genere, quella di qualsiasi comando, basta premere Ctrl-C (ovvero il tasto «c» tenendo premuto il tasto Control).

Anche con Type possiamo usare delle opzioni:
Type FROM File1 TO File2
Crea un file di nome File 2 e vi scrive il contenuto di File1; in questo caso nulla appare nella finestra del Cli. Le parole «FROM» e «TO» possono sempre essere omesse, ma è comunque bene scriverle, se non altro per non far confusione noi stessi;
Type File.doc opt n
L'opzione «N» fà aggiungere automaticamente il numero di linea al testo in output.
Type Qualsiasi_File opt h
L'opzione «H» è molto interessante perché permette di esaminare un file qualsiasi, quindi anche non di testo.

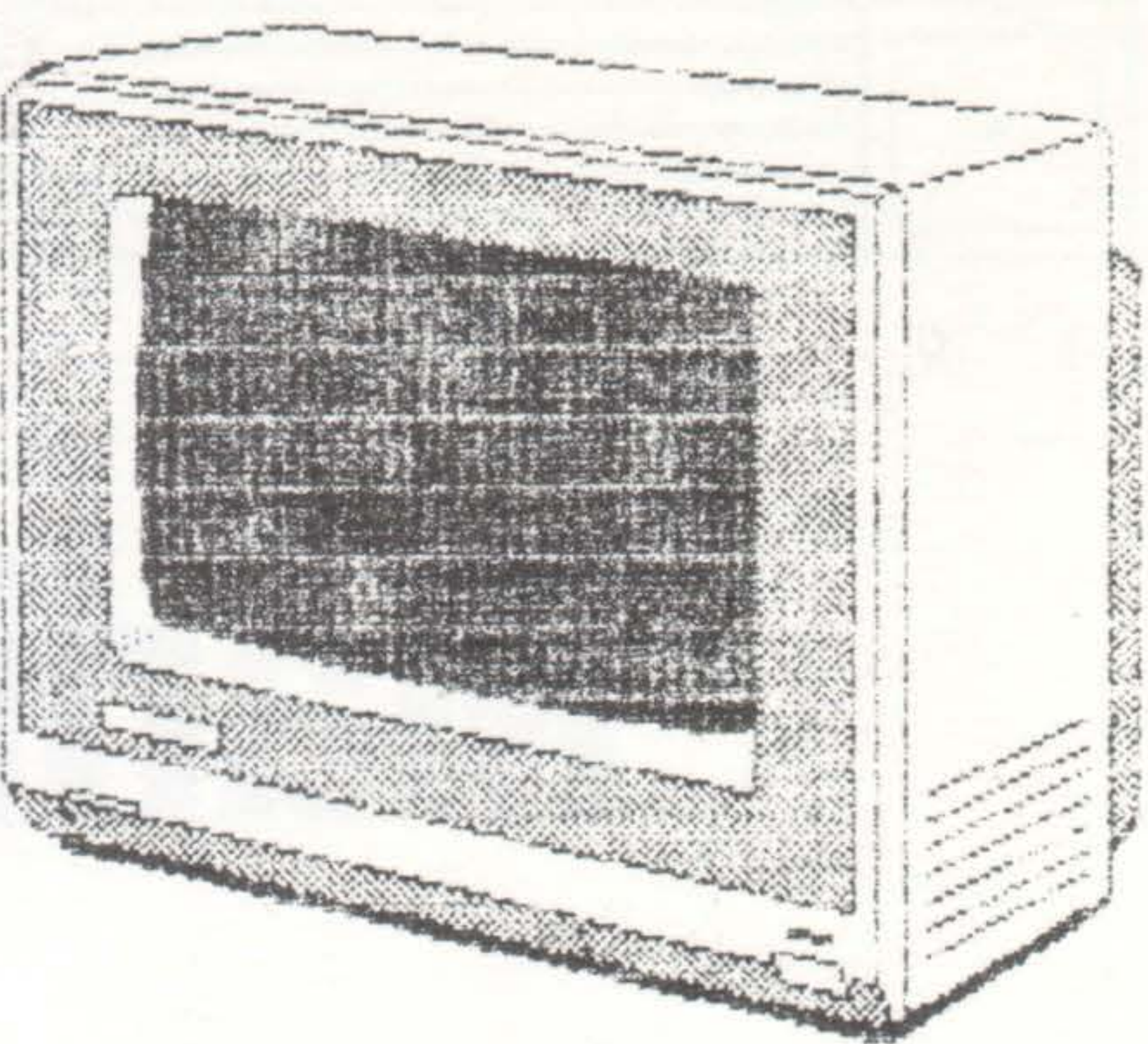
Il formato dell'output in questo caso è un po' particolare, diviso com'è in colonne dal significato differente.

Nella prima colonna abbiamo un numero che indica la posizione del primo byte rappresentato su quella linea, poiché su ogni linea vengono rappresentati 16 byte, la prima colonna della prima linea sarà sempre «0000:» quella della seconda «0010:» (ovvero 16 in esadecimale) e quella della terza «0020:» cioè 32 etc.

Questo semplice comando scriverà in ram il file readme, aggiungendo però ad esso un numero progressivo per ogni linea.

Nella seconda, terza, quarta e quinta colonna troviamo la rappresentazione di sedici byte consecutivi nei quali di ognuno di essi viene scritto il valore in formato esadecimale (che consta di due cifre, potendo un byte avere valori da «00», cioè 0, a «FF» cioè 255).

Nella sesta colonna infine troviamo la rappresentazione dei medesimi sedici byte in forma di carattere ASCII, quando possibile, o altrimenti rappresentati da un punto, quando il loro valore non corrisponda ad alcun carattere ASCII.

Tale rappresentazione è utile per individuare a colpo d'occhio quando una sequenza di byte rappresenta un messaggio o il nome, ad esempio, di qualche libreria.

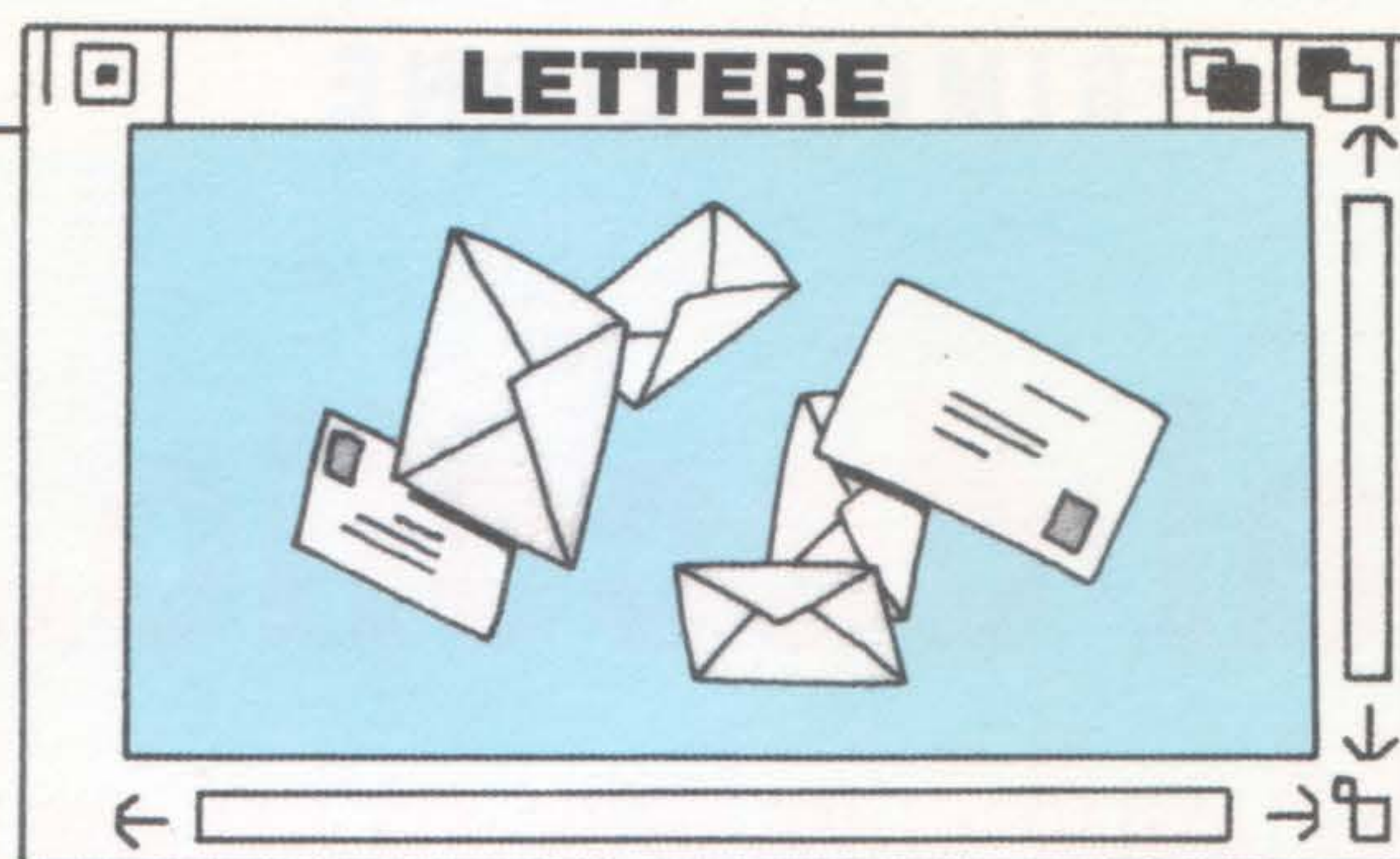
Anche qui, opzioni e percorsi possono essere liberamente combinati, così:
Type From df1:Tools/Readme

To Ram:Readme.bis opt N
riscriverà in Ram il file Readme presente nel cassetto Tools del disco nel drive 1, chiamando il nuovo file «Readme.bis» ed aggiungendo ad esso un numero progressivo per ogni linea.

Quando si usa il comando Type (senza l'opzione «H») su un file che contenga anche caratteri non ASCII quali i caratteri di controllo, questi ultimi possono essere interpretati correttamente (ad esempio il carattere di tabulazione, il cambio di stile, la cancellazione dello schermo etc.) oppure dar luogo alla visualizzazione di caratteri strani e bizzarri, senza che ciò implichi in qualche modo il malfunzionamento del comando Type o dell'Amiga.

E PER FINIRE...

Alcuni consigli; in genere i files di testo si chiamano «ReadMe» o hanno l'estensione «.doc», che sta per «documento» e, benché non sia obbligatorio, è bene seguire questa convenzione nella creazione dei nostri files di testo, soprattutto se si prevede che possano essere esaminati da altri.

Ogni singolo Word Processor (programma di creazione di testi) possiede i propri caratteri di controllo che solo esso Word Processor è in grado di interpretare, ma offre sempre la possibilità di salvare il testo anche in formato ASCII, opportunità che è sempre meglio sfruttare per poter poi esaminare il testo con il comando Type senza andare incontro agli inconvenienti cui si accennava più sopra.

Infine, tutti i listati dei programmi in «C» (detti *sorgenti*) ed i listati dei programmi Basic salvati in modo ASCII (vedi corso di Basic) sono a tutti gli effetti dei testi, pertanto possono essere esaminati con il comando Type.
Anche per questi comandi il consiglio è di sperimentare liberamente, fate solo attenzione alla forma:
Type From A To B
poiché, se esiste già un file con nome «B», esso viene sostituito dal nuovo perdendosi irrimediabilmente. ■

SUONO E TV MONITOR

Il mio Amiga 500 è collegato ad un Tv Monitor Hitachi (CTV 1444): la qualità dell'immagine è buona, ma non posso dire lo stesso per quanto riguarda l'audio. Questo infatti è debolissimo, a volte persino quasi inudibile, e comunque sembra anche distorto: non esistono un amplificatore speciale o un adattatore che mi permettano di elevare il livello di uscita audio del mio Amiga?

Aldo Risici - Alessandria

A dire la verità sembra che il problema possa risiedere più in qualche guasto occorso a Paula (il chip che si occupa, tra l'altro, della gestione del suono) che a guai relativi all'interfacciamento con il tuo Tv-Monitor.

Puoi comunque fare una semplice prova: se hai un amico che possiede un monitor 1081 o un 1084 (quelli in dotazione ad Amiga) prova a collegare il tuo computer ad uno di essi; oppure, connettilo ad un impianto stereo (se ce l'hai) o ad un apparecchio portatile che preveda un ingresso «aux» esterno con i connettori RCA, come quelli audio di Amiga.

Se tutto dovesse funzionare perfettamente allora vuol dire che il tuo Amiga non ha nulla: è il tuo televisore che forse ha un'impedenza di ingresso bassa (sotto i 600 ohms), mentre l'audio del nostro computer deve entrare in un ingresso ad alta impedenza (sopra i 2 Kohms). Potresti dunque utilizzare un preamplificatore o, soluzione più economica, un semplice adattatore di impedenza che troverai in qualsiasi negozio di elettronica.

QUESITI SULLA STARTUP-SEQUENCE

Sento spesso parlare della Startup-Sequence come di una cosa che deve essere modificata di frequente per far funzionare meglio il computer. Cosa si nasconde in questo onnipresente programma?

Massimo Lerici - Rapallo (GE)

Questa volta non ci hai proprio azzeccato! La Startup-Sequence, come dice il nome stesso, non è un programma, ma una sequenza di comandi dell'AmigaDOS (uno «script file») presente nella directory «S» (che sta, appunto, per «sequences»). Il fatto poi che la sequenza sia di «Startup» indica che questa è la prima cosa che viene eseguita all'accensione della macchina. È quindi ovvio che ogni sua modifica cambia la configurazione della macchina.

PAD NUMERICO INCOMPLETO?

Utilizzo il mio Amiga 2000 con tastiera americana per la contabilità del negozio di mia moglie: quando lavoro con lo spreadsheet o con il data base, mi risulterebbe più comodo poter usufruire delle possibilità offerte dal tastierino numerico, ma in realtà i tasti delle parentesi, l'asterisco, lo slash ed il + vengono ignorati dal computer. Il problema è nel software, cioè nel sistema operativo, o nell'hardware?

Ugo Fasoli - Senigallia (AN)

Il problema ha una soluzione molto semplice: i personal computer di un certo livello prevedono la possibilità di ridefinire la tastiera per adattarla alle versioni nazionalizzate; l'Amiga ovviamente non fà eccezione. I programmi del computer Commodore utilizzano a questo scopo il comando «SetMap» incluso normalmente nella directory System, o in quella C: la tastiera di defualt che viene caricata di solito è la Usa0, la stessa utilizzata sull'Amiga 1000, che come saprai ha un tastierino numerico diverso da quello del 2000 o del 500; se inserirai nella startup-sequence dei tuoi programmi il comando "SetMap Usa1", tutti i tasti del Pad saranno attivati.

Prima, però, controlla che nella directory «devs/keymaps» la Usa1 sia presente.

Se hai qualche problema e vuoi una consulenza rapida telefona in redazione ogni mercoledì pomeriggio al numero 02/706329 dalle 15 alle 18: l'esperto è a tua completa disposizione (non in agosto!!).

PROGRAMMI EDUCATIVI

Ho acquistato un Amiga e, date le sue notevoli possibilità che ho ben presto scoperto, intendo usarlo come «computer totale», anche per l'educazione dei miei bambini.

Esistono in commercio dei validi programmi educativi che loro possano utilizzare facilmente?

Olga Valdini - Siena

Certo, per il nostro computer esistono numerosi programmi educativi a tutti i livelli, dal prescolastico all'universitario, primi fra tutti quelli dell'americana Micro Illusions, reperibili con facilità anche in Italia. Purtroppo però i testi sono generalmente in inglese, almeno per ora: speriamo che magari la Commodore Italiana abbia in mente qualcosa nella nostra lingua.

Dal decollo... all'atterraggio

Grazie alle lezioni di volo precedenti, abbiamo preso confidenza con il nostro aereo e con Flight Simulator II. Siamo ora in grado di decollare e volare, di sbizzarrirci in aria e di atterrare.

di CARLO CATTONI

2ª parte

Più sicuri di noi e del nostro velivolo, con il quale abbiamo familiarizzato, oltre che con lo splendido Flight Simulator II della Sublogic, nella puntata precedente, accingiamoci ora a decollare.

Tornati in testata pista, siamo finalmente pronti al grande passo: i controlli pre-volo, in realtà piuttosto complessi, qui si possono ridurre al centraggio di alettoni e timoni, al controllo della posizione dei flaps (tutti su), delle luci (spente di giorno) e del riscaldamento-carburatore, da usarsi eventualmente per 5 secondi con tempo freddo prima del decollo e dell'atterraggio; normalmente off. Eseguiamo.

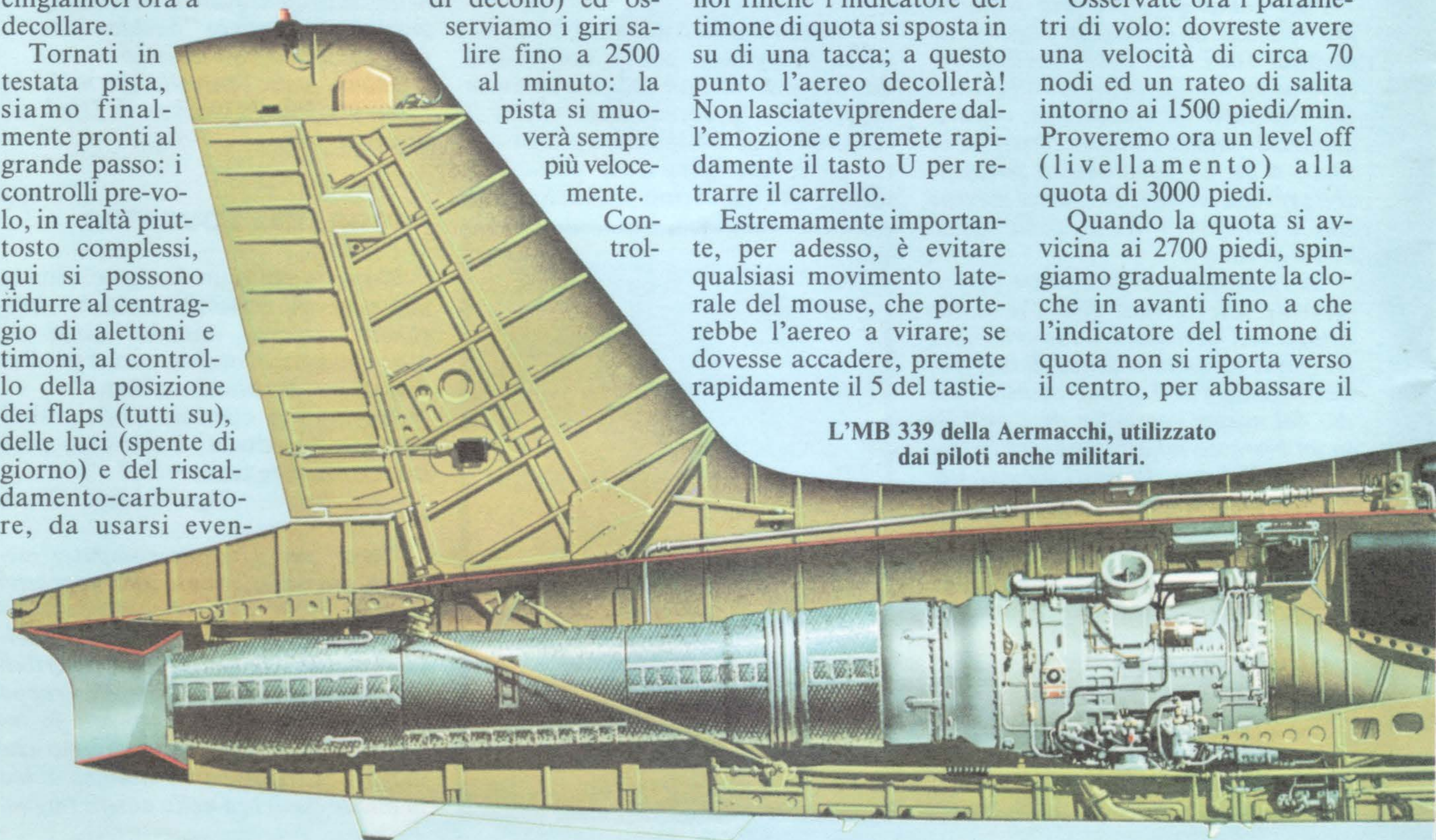
Portiamo con decisione il gas al massimo (potenza di decollo) ed osserviamo i giri salire fino a 2500 al minuto: la pista si muoverà sempre più velocemente. Controlliamo le eventuali piccole deviazioni dell'aereo rispetto alla linea di mezzeria con leggeri spostamenti del mouse. Quando la velocità è prossima ai 55 - 60 nodi, tiriamo la cloche verso di noi finché l'indicatore del timone di quota si sposta in su di una tacca; a questo punto l'aereo decollerà! Non lasciatevi prendere dall'emozione e premete rapidamente il tasto U per retrarre il carrello.

Estremamente importante, per adesso, è evitare qualsiasi movimento laterale del mouse, che porterebbe l'aereo a virare; se dovesse accadere, premete rapidamente il 5 del tastierino numerico (centraggio alettoni).

Premete il tasto B per ottenere una vista della pista che si allontana; poi di nuovo T per la vista anteriore.

Osservate ora i parametri di volo: dovreste avere una velocità di circa 70 nodi ed un rateo di salita intorno ai 1500 piedi/min. Proveremo ora un level off (livellamento) alla quota di 3000 piedi.

Quando la quota si avvicina ai 2700 piedi, spingiamo gradualmente la cloche in avanti fino a che l'indicatore del timone di quota non si riporta verso il centro, per abbassare il

L'MB 339 della Aermacchi, utilizzato dai piloti anche militari.

muso dell'aereo: osserveremo la velocità aumentare.

Cerchiamo intanto di tenere d'occhio l'altimetro, e di non superare eccessivamente la quota di 3000 piedi; gli spostamenti del mouse devono essere sempre graduali e mai violenti, per evitare di incappare nel cosiddetto «overcontrol». Quando la velocità arriva a 110 nodi, portiamo i giri-motore a 1700 ed impegnamoci a mantenere quota e velocità il più possibile costanti per qualche minuto. Dovremmo in breve trovarci su S. Francisco.

Proviamo adesso qualche virata non impegnativa.

Muoviamo il mouse verso destra fino a che l'indicatore di posizione degli alettoni non si sposta di meno di una tacca, e teniamocelo per non più di tre secondi. Riportiamolo poi al centro; noteremo che l'aereo prima si inclina poi, quando centriamo gli alettoni, gradualmente riporta le ali in posizione orizzontale.

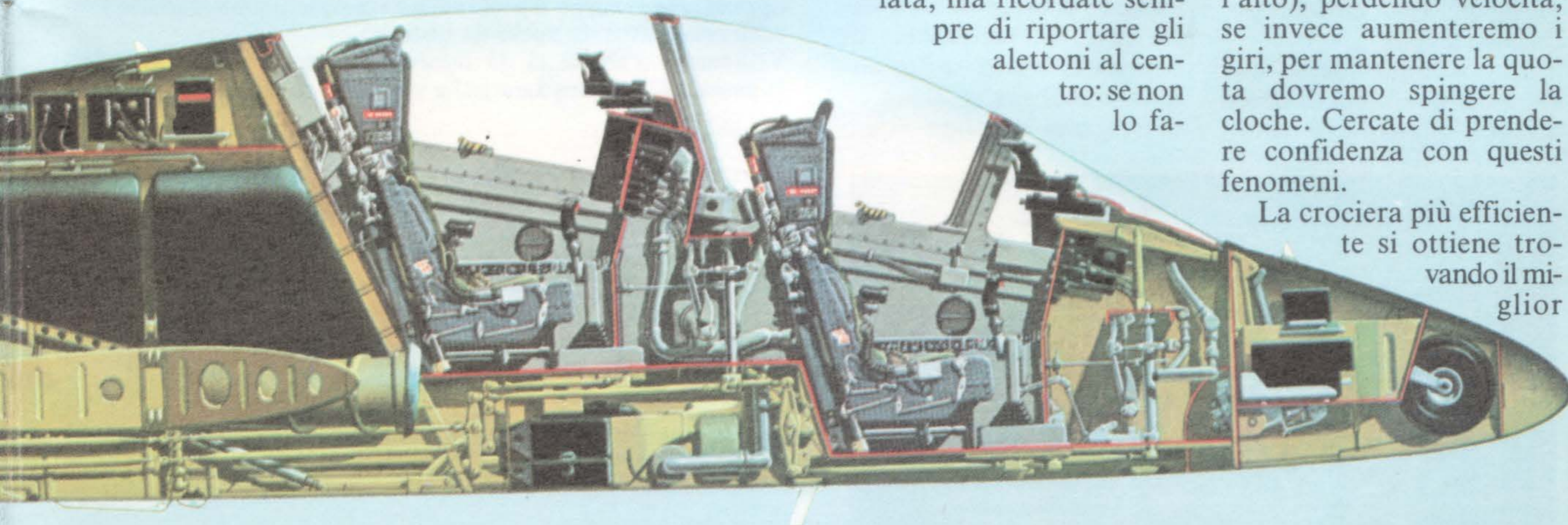
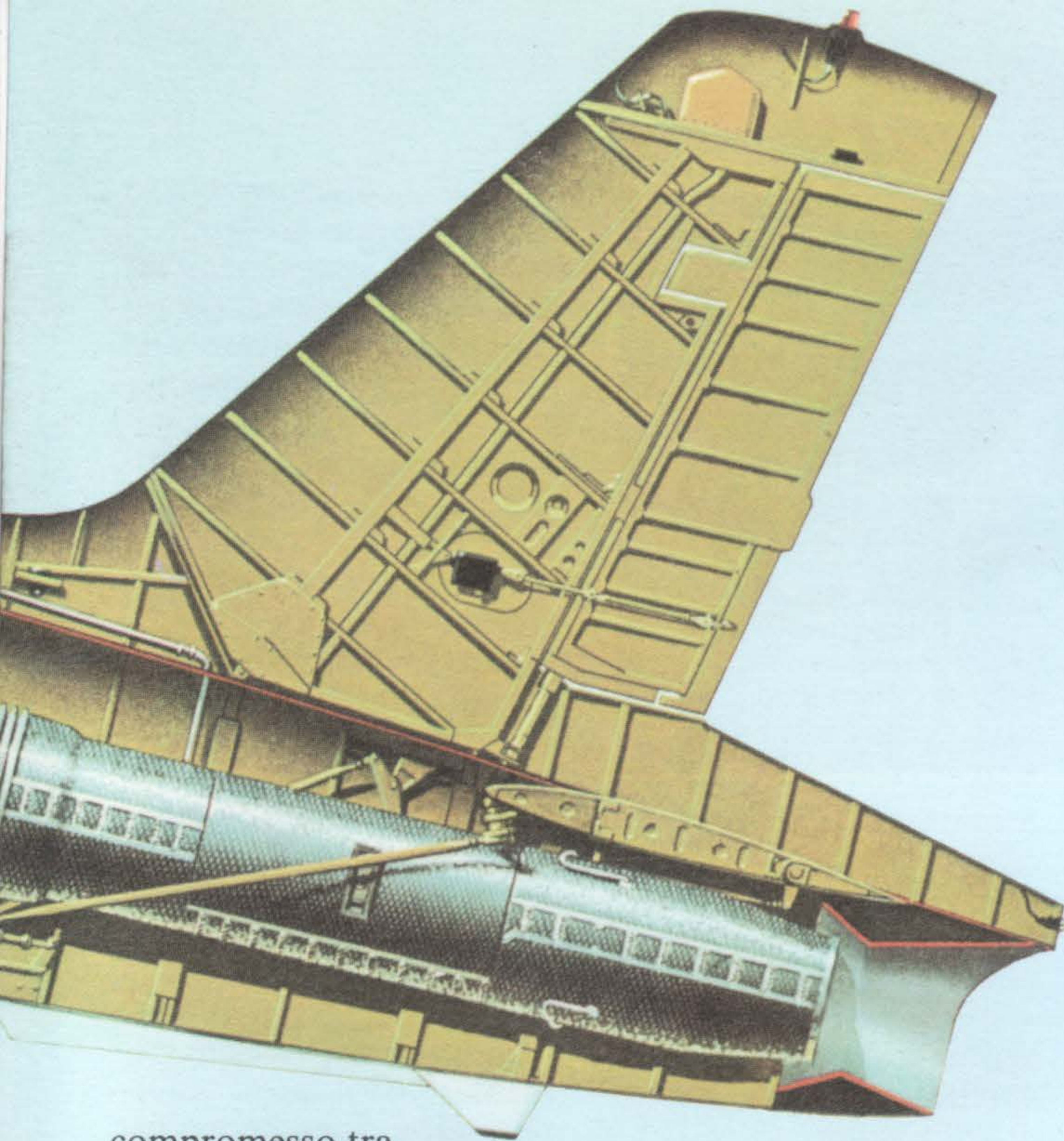
Questo è vero per piccoli angoli di inclinazione sia in virata che in salita o discesa, mentre per angoli più elevati, quando si riportano i comandi al centro, l'aereo rimane in posizione o, addirittura, accentua la manovra (ma, non sono, queste situazioni adatte ai piloti novelli). Fate quindi esperimenti di virata, familiarizzate con le leggere perdite di quota proporzionali alla vostra inclinazione, cercando di mantenervi entro i confini di S. Francisco. Soprattutto ricordate di non esagerare MAI con gli spostamenti del mouse. Assegnatevi ora un compito: supponendo che la vostra bussola segni circa 300 gradi, imponetevi di assumere una prua di 90 gradi, ovvero di andare verso Est.

Impostate quindi una virata a destra facendo in modo che l'orizzonte arrivi ad un'inclinazione di circa 10 gradi. Verificate con l'orizzonte artificiale: sebbene poco visibili, i puntini bianchi sull'arco superiore indicano inclinazioni di 10, 20, 30 e 60 gradi quando sono allineati con la tacca di riferimento a forma di punta rivolta verso il basso. La vostra prua letta sulla bussola comincerà a cambiare e le cifre, una volta arrivate a 359 gradi, torneranno a zero (Nord) e ricominceranno a salire.

Cercate di contenere l'eventuale perdita di quota applicando una leggera tendenza a salire.

Durante questo periodo dovrete controllare l'inclinazione dell'aereo per tenerla costante: date colpi leggeri e ripetuti di mouse quando l'aereo tenderà a tornare in posizione livellata, ma ricordate sempre di riportare gli alettoni al centro: se non lo faceste, essi continuerebbero la loro azione portando il velivolo ad assumere un'inclinazione eccessiva. Quando arrivate ad avere una prua di dieci gradi inferiore a quella desiderata, cominciate ad applicare gli alettoni nella direzione inversa, sempre moderatamente e con dolcezza; gli aggiustamenti fini della direzione si ottengono con piccoli colpi di mouse.

Facciamo qualche esperimento con le variazioni di quota e velocità senza uso del gas; noterete che, se salirete, la velocità diminuirà fino a che il muso dell'aereo tenderà a riabbassarsi e, se viceversa scenderete, la velocità aumenterà ed il muso tenderà a salire. Esercitatevi per qualche minuto. Proviamo invece ora a diminuire il numero dei giri, portandolo a 1500; per mantenere la quota saremo costretti ad assumere un assetto più cabrato (con il muso verso l'alto), perdendo velocità; se invece aumenteremo i giri, per mantenere la quota dovremo spingere la cloche. Cercate di prendere confidenza con questi fenomeni.

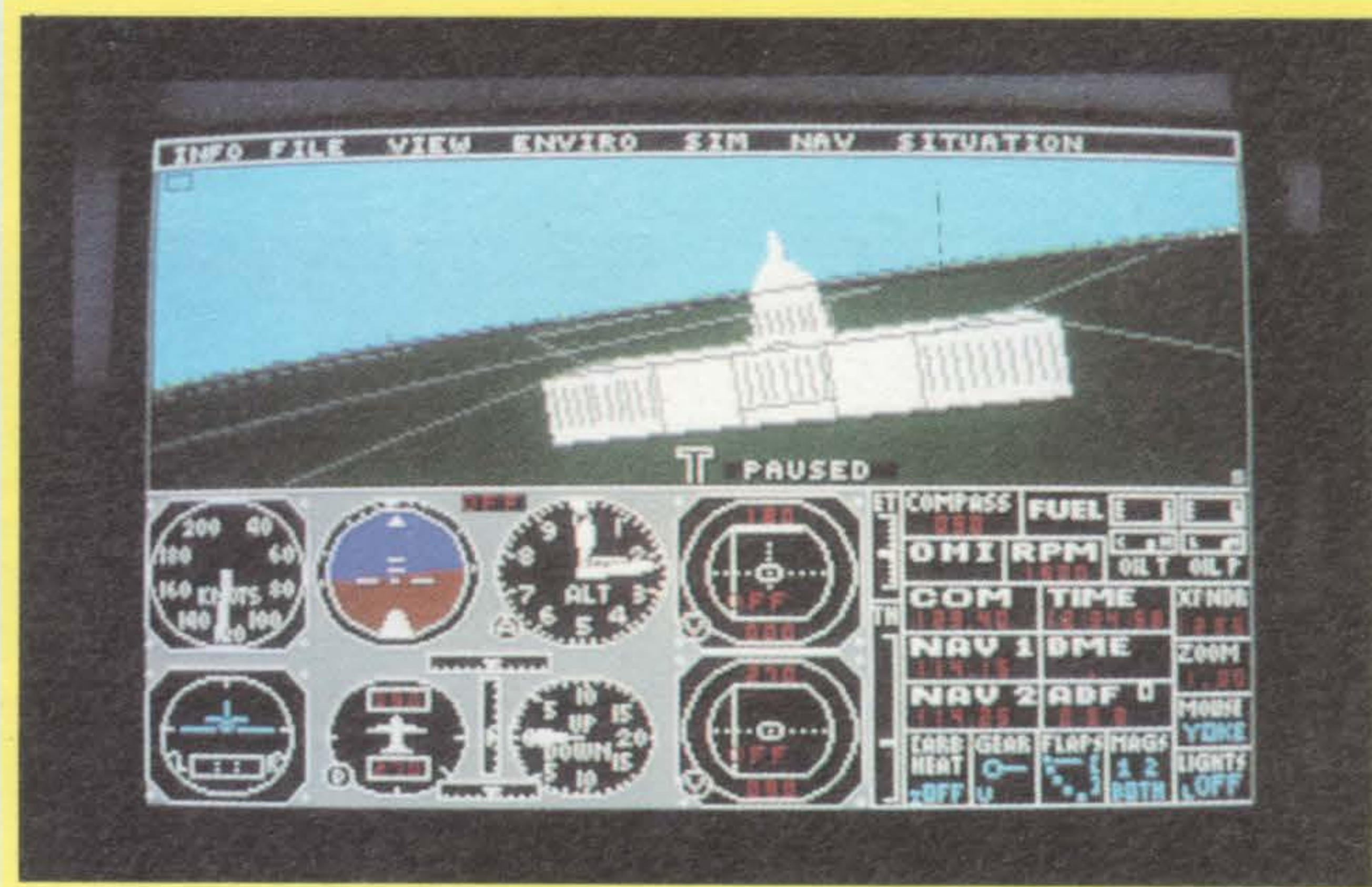
La crociera più efficiente si ottiene trovando il miglior

compromesso tra velocità, giri ed un assetto che non costringa a continue correzioni per mantenersi costante: fate molti esperimenti, ma sempre con calma e senza esagerare con le manovre! Ricordate poi che il regime utile del motore in volo di crociera va dai 1600 ai 1900 giri/min.

Provate ora a cercare di ottenere la migliore velocità di salita: portate i giri a 2200 ed impostate un assetto che vi permetta di salire a 90 nodi e 1200 piedi al minuto: noterete che il timone di quota è quasi centrato. Se poi aumentate il regime fino a 2500 giri/min., otterrete 105 nodi con 1700 piedi al minuto, con il timone lievemente «a picchiare». Fate un po' di pratica.

Impostiamo adesso una discesa; livelliamo l'aereo a 5000 piedi, 120 nodi, 1700 giri/min.; ora togliamo completamente il gas, e manteniamo il velivolo in un assetto che permetta una velocità di 80 nodi: questo sarà l'assetto «a salire».

Proviamo ora ad abbassare i flaps di una posizione (tasto]), cercando di mantenere gli 80 nodi; l'assetto dovrà essere modificato, tendendo ora alla discesa.

Estraiamo i flaps di un'ulteriore posizione (osservare l'indicatore): per mantenere gli 80 nodi dovremo dare cloche-avanti tendendo a picchiare; esercitatevi a controllare le discese, imponendovi una velocità da mantenere costante; provate pure ad estrarre il carrello e ad osservare le variazioni di comportamento utilizzando anche il gas.

Eseguite ora una «riattaccata» (manovra di scampo): partite dalla situazione «via il gas — flap fuori — carrello fuori — assetto a scendere» tipica di una discesa da atterraggio. Date quindi tutto gas, togliete rapidamente il carrello e la prima tacca di flaps; dopo 10 secondi, togliete anche la seconda e regolarizzate il volo. Questa manovra vi tornerà molto utile quando vi succederà di dover rinunciare all'atterraggio all'ultimo istante.

Provate adesso ad eseguire virate più «tirate» e ad esplorare limiti più elevati; se vi dovesse capitare di compiere un errore o di voler rivedere una situazione interessante, scegliete «instant replay» dal menù «situation» e godetevi alla moviola gli ultimi 75 secondi del vostro volo, magari usando l'opzione «spot». Fate uso frequente di questa possibilità. Usate anche, in tutti i vostri voli di prova, l'opzione «save to ram», sempre dal menù «situation»: permette di memorizzare situazioni intermedie, che potrete richiamare in seguito con il «recall», o salvare su disco. Seguite sempre le indicazioni fornite dal programma.

Provate anche le altre possibilità di comandare l'aereo, mediante la sola tastiera o con il joystick; la combinazione tastiera — mouse è la più indicata, ma potreste preferire diversamente.

Siamo finalmente giunti al momento di eseguire un volo completo, da Oakland all'aereoporto di Livermore Municipal, situato ad una distanza di circa 20 miglia, ad una quota di 397 piedi (vedere la carta di navigazione di S. Francisco).

Tornate dunque alla situazione iniziale; portatevi alla testata-pista opposta alla 27R, ovvero alla 9L (9 sta per 90 gradi di orien-

TUTTE LE OPZIONI

I menù a disposizione ci permettono di selezionare una marea di opzioni, che rendono il programma «completissimo» (alcune possono essere attivate anche da tastiera).

Vediamole: l'INFO ci dà informazioni sugli aerei disponibili (Cessna 182 e Gates Learjet) e sugli autori; il FILE ci permette

di scegliere l'aereo (attenzione però: il manuale stesso ci avverte che la simulazione del jet è meno accurata) anche mentre siamo

tamento.
Punta quindi ad est) e giratevi orientando la prua verso il monte che appare all'orizzonte.

Dopo aver eseguito i controlli pre-volo decollate e, immediatamente dopo aver staccato le ruote da terra, virate per prua 80 gradi. Ricordatevi il carrello!

Ottenete un level-off a 3000 piedi, con i parametri di crociera 1700 giri/min e circa 110 nodi.

Alla vostra destra passerà S. Francisco; guardatevi attorno con le «viste», ma rammentate che il controllo del velivolo ha priorità assoluta anche rispetto al mantenimento dei parametri e, a maggior ragione, rispetto al «lato turistico»!

Insomma, non si deve rischiare di precipitare cercando in ogni modo di assumere una determinata prua od un certo rateo di discesa, se questo impone al velivolo assetti impossibili da mantenere.

A questo punto dovreste avere davanti a voi il monte e tre cittadine e, sulla sinistra del monte, due strade che si incrociano, una delle quali va più o meno nella vostra stessa direzione. Usate l'incrocio come riferimento e puntate su di esso; la vostra prua dovrà essere variata di pochi gradi. Vi allenerete così ad utilizzare punti di riferimento al suolo, indispensabili nella navigazione a vista.

Quando sarete in prossimità dell'incrocio, virate di nuovo per prua 80 mantenendovi paralleli alla strada leggermente alla sinistra dell'incrocio; come riferimento, la strada dovrà scorrere all'altezza dell'indicatore dell'autopilota. Punterete dunque verso la cittadina che appare all'orizzonte diritta davanti a voi, Livermore.

Ad un certo punto il drive entrerà in funzione e caricherà i dati della zona, compresa l'immagine della pista, che vi apparirà come una striscia grigia praticamente parallela alla strada; sarete quasi allineati con essa.

Questo è il momento di prepararsi all'atterraggio. Impostate una discesa di avvicinamento: motore a 1000 giri/min., una tacca di flaps; mantenete una velocità di 90 nodi e cercate di curare l'allineamento con la pista usando il Marker (la T che avete supra il cruscotto); la pista dovrà risultare allineata ed il più possibile parallela ad esso, quindi verticale. L'allineamento dovrà essere perseguito con piccolissimi e progressivi spostamenti del mouse, per non incorrere in virate vere e proprie che rischierebbero eccessive correzioni; tenete presente che questi sono assetti abbastanza critici per un aereo, e che la manovrabilità ne risulta limitata. La vostra rotta dovrà passare comunque dagli 80 ai 73-74 gradi di quando poserete le ruote a terra. Giunti a 2400 piedi togliete il gas, date una seconda tacca di flaps ed estraete il carrello; mantenete i 70 nodi in discesa; il vostro assetto sarà notevolmente «a picchiare». Curate l'allineamento. Passata la quota di 600 piedi (200 dal suolo dell'aereoporto, che si trova a 400 piedi circa sul livello del mare) cominciate la richiamata tirando la cloche verso di voi; dovreste essere praticamente sulla testata-pista. Scopo dell'atterraggio è arrivare a pochi centimetri da terra e «stal-

(segue a pag. 46)

in volo, oppure di attivare le demo o, ancora, di entrare nel modo combattimento (WW1 ace) o nel modo multi-player via modem o cavo. Con il VIEW possiamo scegliere le «viste» e settare da sottomenù i parametri dell'ipotetico aereo spot («chase plane»); in ENVIRO sceglieremo la stagione in cui volare, ed i sottomenù per creare nubi, vento e nebbia. L'ora del giorno va cambiata clickando direttamente sull'orologio del pannello strumenti. Con SIM possiamo adeguare i parametri della simulazione a nostro piacimento: modi del cursore, sensibilità dei comandi, livelli di realismo e di affidabilità dell'aereo.

Il menù NAV è molto importante in quanto ci consente, con POSITION SET e le coordinate stampate sulle carte nautiche, di portarci in qualsiasi parte del «mondo» dell'FSII; inoltre vi si trovano i sottomenù mappe, autopilota e «slew» (puntamento rapido). Nel SITUATION, infine, troviamo la possibilità di caricare le situazioni preregistrate, di salvarne di nostre su ram ed in seguito eventualmente su disco, e di richiamarle a volontà. È presente anche l'ISTANT REPLAY con il quale rivedere gli ultimi istanti del nostro volo, che ci mette a disposizione un tempo di registrazione massimo di 75 secondi (possiamo però settare un tempo inferiore).

L'uso dei menù risulta sempre molto facile ed intuitivo.

SUGGERIMENTI E TRUCCHI VARI

Una delle opzioni meno conosciute (ma non per questo meno utili) del comando List è rappresentata dalla parola chiave «keys»: questa, battuta dopo l'istruzione, farà apparire a fianco ai files elencati, anche il numero del blocco del File Header (cioè quello del primo settore occupato) o quello della Directory, incluso in parentesi quadre per distinguerlo dalle dimensioni in bytes. Utile quindi se si vuole esplorare un disco. Un esempio del comando è:
LIST DF1: KEYS
Ricordarsi di inserire lo spazio.

Anche se i manuali non lo riportano, è possibile utilizzare il conosciuto spreadsheet Analyze! ed il programma di telecomunicazioni Online! in risoluzione interallacciata, richiamandoli semplicemente da Cli con il comando:
Run Online! * (oppure Analyze! *).

Questo risulta molto utile soprattutto in Analyze!, in quanto si può avere direttamente sullo schermo il doppio delle celle normalmente visibili.

Tutti quelli che possiedono un solo drive (speriamo siano sempre di meno!) e che avranno già sperimentato la difficoltà dell'avere una directory di un disco che non sia quello di sistema (bisogna ricordarne il nome esatto), accetteranno di buon grado il consiglio che segue: digitate Dir ? ed il computer risponderà con: DIR,OPT/K:.

Cosa è successo? Il comando Dir è stato caricato da disco, ma il computer aspetta che voi gli comunichiate le opzioni selezionabili, dopo avervele mostrate.

A questo punto inserite il disco del quale volete avere la directory, premete return, ed il gioco è fatto.

Il problema del flickering è forse quello che più cruccia gli Amighi. È possibile però ridurlo al minimo ed utilizzare l'interlacciamento anche nel workbench: il segreto è nella scelta dei colori, che dovrebbero essere poco contrastati ma nello stesso tempo ben visibili.

Una buona combinazione si ottiene così: entrate nelle Preferences, dove troverete il conosciuto angolino destinato al controllo dei colori, con i quattro registri degli stessi e gli «slider» marcati R, G e B.

Utilizzate ora questo schema:

1	2	3	4
R G B	R G B	R G B	R G B
0 4 6	0 0 0	0 0 9	15 0 0

La procedura da seguire è questa: selezionato il primo registro, portate i cursori a zero (tutti a sinistra, colore nero) e clickate nel lato destro di ogni barra R, G e B per il numero delle volte specificato nella tabella.

Il risultato potrà forse non piacervi dal punto di vista estetico, ma conviene provare!

Turbizziamo la startup-sequence del nostro Workbench con alcuni semplici suggerimenti: innanzitutto eliminando tutte le linee di output video (echo, type etc.) che richiedono il caricamento del comando e tutte le coppie IF ENDIF che verificano l'esistenza di file o di directory. Poiché già sappiamo se queste ci sono o meno, è inutile far eseguire il controllo ogni volta. Se si possiede un'espansione di memoria aggiungete queste istruzioni:
C:Addbuffers df0: 30
C:Addbuffers df1: 30
(solo se si ha il secondo drive). Queste istruzioni riservano 30 cache buffer per ogni drive, diminuendo sensibilmente il numero di accessi al disco. Eliminate l'istruzione Binddrivers se non usate hardware aggiuntivo o ramdisk particolari; specificate SEMPRE l'intero path (percorso) per giungere ai comandi, dato che il sistema operativo li cerca prima nella directory corrente e poi nella C, dove sono

normalmente. Per esempio, al posto di RUN EDITOR usate C:RUN :TOOLS/EDITOR ammesso che l'editor sia nella directory tools. Quando avete terminato di fare tutte le modifiche al vostro dischetto da combattimento, eseguite quanto segue:
COPY nome_sorgente TO nome_destinazione ALL QUIET che copierà l'intero disco file per file e non settore per settore come fà il diskcopy. Il vantaggio è che il sistema operativo riorganizzerà nella maniera migliore tutti i files, anche se per questo lavoro il top è costituito dal DiskArranger, pubblicato sul primo fascicolo della nostra rivista.

Modem maniaci, queste righe fanno per voi! Avrete certamente sentito

parlare, anzi lo userete sempre, del programma Arc: questo, più che un programma, è diventato uno standard per la compattazione dei files, poiché meno tempo si rimane collegati con una bbs, meno soldi si regalano a mamma Sip. I file «arcati» devono, per essere usati, venire appunto «dearcati»; per far questo è sufficiente dare un solo parametro ad Arc: mettete questo breve tool nella directory C, in modo da poterlo usare come un normale comando Dos, e portatevi nella directory nella quale volete sia scritto il file definitivo.

Supponendo che questa fosse DEARCATI, e che il file compattato si chiami ESEMPIO.ARC e si trovi nel drive df1, digitate:

cd df0:dearcati
arc x df1:esempio

La X è il prefisso che indica di scompattare tutti i files contenuti in ESEMPIO (notare che non è necessario aggiungere l'estensione .arc), mentre per avere un elenco completo delle opzioni disponibili dovete dare:

arc h

Ogni tanto è bene far riposare il nostro povero computer, piuttosto che costringerlo a saltare in continuazione quando non ce n'è bisogno...

Gli amanti del Basic avranno letto e visto centinaia di volte una linea di programma come questa:

120 IF MOUSE(0) = 0 THEN 120

che testa la pressione dei tasti del mouse stesso. Non c'è nulla di errato ma, visto che l'Amiga è un computer multitasking, (il che in parole povere significa che il «cpu time» viene suddiviso tra i processi attivi a seconda della richiesta), cerchiamo di non appesantire inutilmente il processore. Apparentemente quella linea 120 sembra non chiedere troppo lavoro al 68000, ma, invece, ne mangia parecchio! Al suo posto è meglio usare, per un migliore sfruttamento del multitasking, l'istruzione SLEEP che «addormenta» il computer finché qualcosa non lo desta:

120 IF MOUSE(0)=0 THEN SLEEP

Potreste aver bisogno di scrivere direttamente qualche riga sulla stampante, ma di non aver voglia di caricare un wp e di non voler scriere utto il file prima di mandarlo in stampa. Usare allora questa semplice riga di comando:

COPY * TO PRT:

Da questo momento, alla pressione di Return, tutto ciò che avrete scritto sullo schermo verrà stampato tale e quale e potrete utilizzare in pratica la vostra stampante come fosse una comune macchine da scrivere.

Per uscire premete Ctrl-C.

Vi siete creati un disco CLI e vorreste avere l'orologio sempre in vista (meglio evitare di perdere appuntamenti importanti...): inserite nella startup-sequence (digitando il comando ED s/startup-sequence) i seguenti comandi:

DATE ?
NEWCLI
CLOCK

Salvate le modifiche battendo Esc poi X e return.

Se avete già il comando Date nella sequenza, non è necessario aggiungerlo; selezionate quindi l'orologio digitale, posizionatelo e attivate la finestra Cli 2.

Il nome della nuova RamDisk del Workbench 1.3 resistente al reset ed a qualsiasi Guru è RD0: la sua peculiarità è che, se usata con il Kickstart 1.3, permette di effettuare il bootstrap dell'Amiga dalla ram stessa!

Digitate:

TYPE DEVS:MOUNTLIST

ed assicuratevi che RD0: sia presente nel file Mountlist e che HighCyl sia settato a 79; in caso contrario editate e correggete voi stessi. Battete ora:

MOUNT RD0:
DISKCOPY DF0: TO RD0:

ed inserire il Workbench 1.3 nel drive interno. A copia finita resettate, provate a togliere tutti i dischetti dai drives e dopo qualche secondo avrete già caricato il Workbench precedentemente copiato in Ram. Davvero troppo utile.

Può capitare di cancellare un file per errore e sappiamo tutti cosa questo significhi: ore di lavoro buttate al vento! Esiste però una semplice possibilità di recuperare un file senza ricorrere a complicate utility presenti in commercio, usando un comando del Dos: il Diskdoctor. Come certamente saprete, la sua funzione è quella di «aggiustare» i file rovinati, o per lo meno di cercare di ricucire gli errori trovati sul dischetto. Se lanciato IMMEDIATAMENTE dopo l'errata cancellazione del file in questione, potrebbe farlo tornare in vita; diciamo potrebbe perché non funziona sempre, poiché non è stato pensato per questo genere di problemi. Probabilmente è perché il Diskdoctor va a scrivere su dei settori che appartenevano al file cancellato, che in questo modo sarà definitivamente perso. In ogni caso, la cosa più importante è che non dovete far scrivere ASSOLUTAMENTE niente dopo aver cancellato per errore il vostro file.

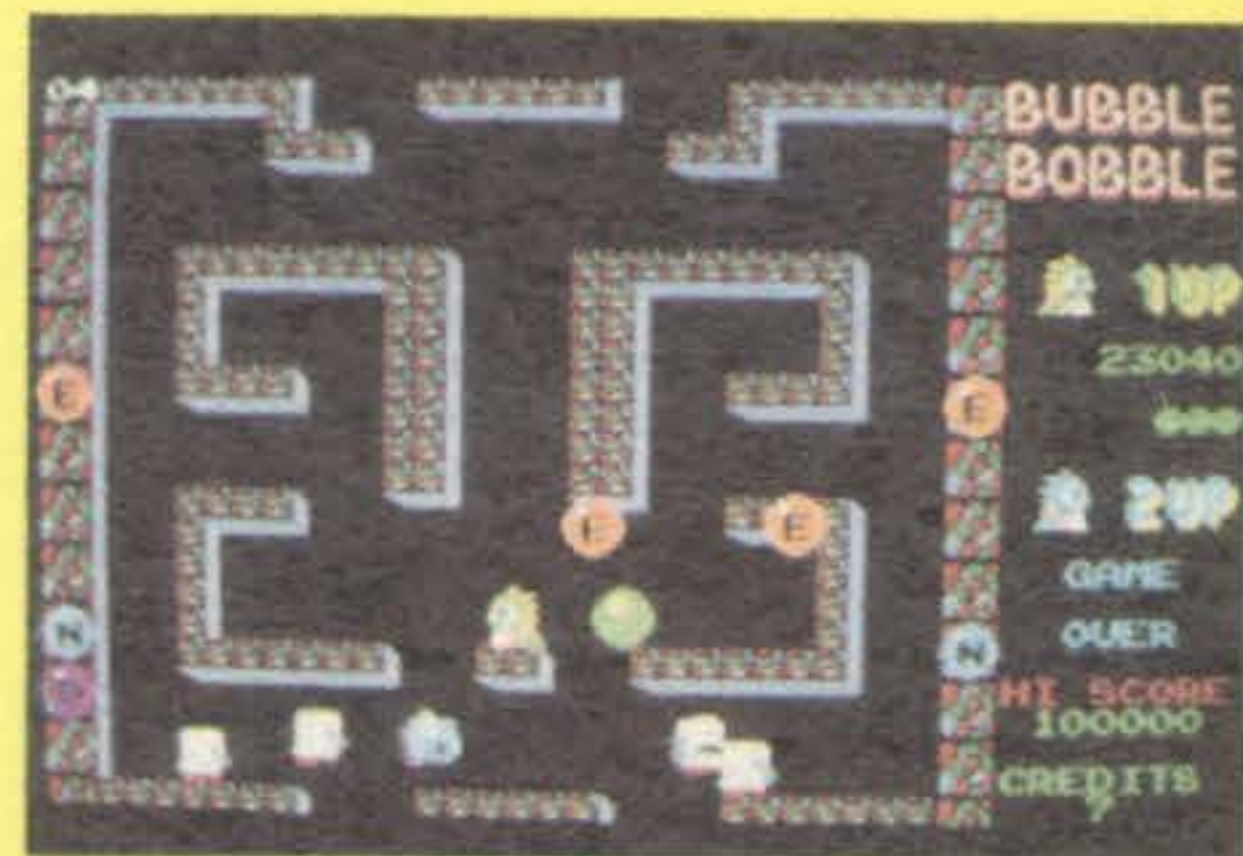
POKE CORNER

Inauguriamo da questo numero di Amiga Byte l'angolo delle poke, riservato a tutti quelli che passano ore ed ore a distruggere il joystick nel vano tentativo di terminare un gioco. Trovare i numeri magici nei programmi Amiga è senza dubbio un'impresa più ardua che su altri computer, ma ci siamo riusciti. Per inserire le vite infinite nei vari giochi che proporremo dovrete SEMPRE usare il programma NEWZAP presente nel dischetto allegato a questo stesso fascicolo: battete come prima cosa il nome del file (scritto nella tabella qui riportata) e successivamente i codici da trovare preceduti dal simbolo del dollaro; una volta scovati, questi ultimi verranno evidenziati. Portatevi con il puntatore sopra la prima cifra e clickate: entrerete così in modo edit esadecimale. Ora digitate i codici CORRISPONDENTI che trovate nella tabella ed il gioco è fatto! Può capitare che non riusciate a trovare le stringhe esadecimali (di rado, ma accade) e ciò significa che la versione del programma in vostro possesso è differente da quella usata da noi. Ovviamente, dopo ogni modifica clickate sul SAVE, il che implica che è bene lavorare su una copia del gioco.

POKE CORNER by Galinberti

CODICI DA CERCARE	CODICI DA INSERIRE
Sidewinder	file "SIDEWINDER" lunghezza 31844
52 68 00 08 48 e7	4e 71 4e 71 48 e7
53 68 00 08	4e 71 4e 71
Bubble Bobble	file "BUBBLE"
61 00 00 b8 61 00 2f e8	4e 71 4e 71 61 00 2f e8
61 00 00 44 61 00 2f a6	4e 71 4e 71 61 00 2f a6
Into the Eagle's Nest	file "E0"
64 00 07 14	4e 71 4e 71
81 01 33 c0 00 00 0f de	4e 71 33 c0 00 00 0f de
Joe Blade	file "JOE"
65 00 f6 80 04 39	65 00 f6 80 60 06

1> ▮

A destra, le foto di «Sidewinder» (in alto) e di «Bubble Bobble».

Un modem per amigo

Un mondo in rapida espansione, quello della telematica. Per necessità o per diletto, ricevere e trasmettere dati via telefono è il sistema più attuale per comunicare. Come farlo con Amiga.

di VITTORIO FERRAGUTI

Per comunicare con Amiga sono necessari, molto semplicemente, un programma per comunicazione ed un modem.

La scelta di un modem adatto alle proprie esigenze è importante ed è anche l'ostacolo principale contro il quale si scontrano i meno esperti. Val la pena ricordare che questo dispositivo, il cui nome deriva dalla fusione dei termini «MOdulatore» e «DEModulatore», ha lo scopo di convertire i segnali digitali provenienti dalla porta seriale del computer in segnali analogici da trasmettere per mezzo della linea telefonica e viceversa.

La differenza principale tra i diversi tipi di modem è costituita dalla velocità di trasmissione, misurata in baud (bit trasmessi al secondo). In genere le velocità disponibili sui modem ad uso amatoriale, ovvero di costo inferiore alle 500.000 lire, sono due: 300 baud o 1200 baud. In alcuni casi è possibile trovare, sostituita al termine baud, la sigla equivalente BPS (dall'inglese «Bits per second»).

Possiamo effettuare una suddivisione dei modem in base alla loro velocità ed

allo standard di comunicazione; sempre rimanendo nell'ambito hobbystico, le categorie principali da considerare sono tre:

- modem V21 (300 baud FULL duplex)
- modem V22 (1200 baud FULL duplex)
- modem V23 (1200 baud HALF duplex)

La prima sigla indica a quale standard di comunicazione europeo (secondo le specifiche CCITT) è conforme il modem: la dicitura «full» o «half» duplex indica la capacità del modem di trasmettere o meno segnali mentre contemporaneamente ne sta ricevendo.

In genere questi standard sono compatibili verso il basso: ciò significa che un modem V22 o V23 generalmente è in grado di trasmettere anche secondo lo standard V21.

Molti acquirenti sono tratti in inganno dalla dicitura «1200 baud» sulla confezione del modem, e non controllano a quale standard di comunicazione esso sia conforme. All'atto pratico tuttavia c'è una differenza enorme tra gli standard V22 e V23: pur trasmettendo i dati alla stessa velocità, essi sono infatti totalmente incompatibili.

Mentre il V22 è adottato dalla maggior parte dei sistemi remoti che trasmettono a 1200 baud, quali ad esempio tutti i Bulletin Board Systems (BBS) operanti sul territorio nazionale o la rete nazionale a commutazione di pacchet-

to (ITAPAC), lo standard V23 viene utilizzato in Italia quasi esclusivamente per il servizio Videotel.

La scelta del modem deve essere fatta dunque in base alle proprie esigenze e naturalmente alla somma a disposizione per l'acquisto. I modem a 1200 baud, essendo più veloci, consentono un notevole risparmio di tempo in quanto la trasmissione di un file o di un testo avviene più rapidamente; un risparmio che si traduce anche in termini monetari, visto che la durata dei collegamenti influisce pesantemente sulla bolletta telefonica, soprattutto in teleselezione. I modem a 300 baud invece offrono il vantaggio di essere molto più economici ed affidabili, poiché la velocità di trasmissione è proporzionale alla quantità di disturbi presenti sulla linea telefonica.

Nel caso in cui il modem serva solo per collegarsi a sistemi informativi (banche dati, archivi da consultare) non è necessaria un'elevata velocità di trasmissione: qualora si debbano invece trasferire file o programmi, è consigliabile un modem più veloce.

AMIGA WORLD COURTESY

I COMPATIBILI HAYES

L'ultimo fattore da prendere in considerazione al momento di scegliere il modem da utilizzare con Amiga è la cosiddetta compatibilità Hayes. Essa consiste nella presenza di un set di comandi esteso, che prende il nome dall'azienda americana che lo ha ideato ed implementato per la prima volta sui propri modem, con il quale è possibile settare via software tutte le funzioni desiderate (velocità, duplex, chiamata e risposta automatiche etc.). Generalmente i programmi di comunicazioni richiedono la presenza di un modem Hayes-compatibile per poter utilizzare tutte le opzioni più avanzate, come la chiamata automatica o il riconoscimento del segnale di occupato.

Il software di comunicazione riveste in ogni caso una grande importanza per l'utilizzo corretto del modem; è inutile infatti dotarsi di apparecchiature sofisticate e costose se il programma utilizzato per pilotarle non è all'altezza. È importante verificare la presenza di alcune caratteristiche al momento della scelta; le più rilevanti generalmente sono:

- Auto Redial: attivando questa funzione il modem compone automaticamente il numero telefonico indicato, e continua a chiamarlo ad intervalli regolari fino a quando non riesce ad effettuare la connessione. Alcuni programmi permettono di inserire più numeri telefonici, che saranno chiamati in sequenza.
- Macro Keys: per risparmiare tempo di collegamento ed errori di battitura è possibile ridefinire i tasti funzione di Amiga, assegnando ad ognuno una stringa di caratteri (ad esempio il proprio nome o il codice di accesso ad un sistema) e salvandola su disco. Una volta collegati sarà sufficiente premere un tasto per trasmettere l'intera stringa o sequenza di comandi memorizzata.
- Script Files: alcuni programmi più sofisticati prevedono la possibilità di preparare intere sequenze di comandi, in grado di rendere completamente automatica la procedura di collegamento ad un sistema remoto, utilizzando uno speciale linguaggio di programmazione. Questi «script files» (denominati anche «command files») possono ad esempio ordinare al computer di effettuare una chiamata ad un numero telefonico fino a che non risponde e di inserire automaticamente il vostro nome e la vostra password non appena collegati.
- Emulazione di terminali: in genere i sistemi remoti ai quali è possibile accedere via modem si limitano ad inviare delle semplici stringhe di caratteri, che scorrono sul monitor come se venissero stampate su carta da una telescrivente. In alcuni casi però è prevista la possibilità di utilizzare terminali più sofisticati, che permettono di sfruttare l'intero schermo come fosse una pagina,

Il sistema professionale Videotex, uno standard utilizzato dai «cugini» francesi. Da un terminale in stazione o in aereoporto si accede a mille informazioni.

I NODI OPUS ITALIA

Node	Board's Name	Phone Number	City	Baud
33	*Italy*	*39-971-35447*	*I*	*1200*
1	*Fido Pz*	*39-971-35447*	*Potenza Italy*	*1200*
2	*Fido PEG*	*39-971-35447*	*Potenza italy*	*1200*
331	*NorthernC*	*39-2-2666502*	*Italy*	*2400*
1	*Ipotesi*	*39-2-2666502*	*Milano Italy*	*2400*
2	*Fido Mi*	*39-2-228467*	*Milano Italy*	*1200*
3	*D.M.B.*	*39-2-315754*	*Milano Italy*	*1200*
4	*Prisma*	*39-372-436900*	*Cremona Italy*	*1200*
6	*NEMO*	*39-2-8245137*	*Rozzano Italy*	*1200*
7	*Blue Net*	*39-321-28929*	*Novara Italy*	*1200*
8	*Epsonlink*	*39-30-9719440*	*Brescia Italy*	*2400*
9	*Opus Va*	*39-331-263425*	*Varese Italy*	*2400*
10	*Euro Elettronica*	*39-373-86966*	*Crema Italy*	*1200*
11	*Amigaline*	*39-30-2420452*	*Brescia Italy*	*2400*
12	*BBS2000*	*39-2-706857*	*Milano Italy*	*1200*
13	*OPUSNOVA*	*39-342-493782*	*Sondrio Italy*	*1200*
14	*Howard The Duck's*	*39-2-6551412*	*Milano Italy*	*1200*
15	*Clessidra New*	*39-2-4159728*	*Milano Italy*	*1200*
16	*Lario Net*	*39-341-735693*	*Mandello L. (Co) Ita*	*1200*
17	*Telesibioc*	*-Unpublished-*	*Milano*	*1200*
332	*Italy88 Telnet*	*39-521-38982*	*Italy*	*2400*
1	*Bit Show*	*39-521-38982*	*Parma Italy*	*2400*
2	*Bit Show 2*	*39-521-285014*	*Parma Italy*	*1200*
3	*C.I. CBS (Fido FI)*	*39-55-253606*	*Scandicci Italy*	*2400*
4	*Digic Link*	*39-55-282365*	*Firenze Italy*	*1200*
5	*Opus Rapallo*	*39-185-274020*	*Rapallo Italy*	*1200*
6	*Genova 2000*	*39-10-3994240*	*Genova Italy*	*2400*
7	*Fido Fo*	*39-541-773527*	*Rimini Italy*	*1200*
9	*OCA System*	*39-51-6343719*	*Bologna Italy*	*2400*
10	*Blue Sea BBS*	*39-10-3770365*	*Genova Italy*	*1200*
11	*Datatel*	*39-59-688994*	*Carpi Italy*	*1200*
12	*Utopia*	*39-573-368164*	*Pistoia Italy*	*1200*
333	*Fri-Ve-Net*	*39-434-32020*	*Italy*	*1200*
1	*Fido Pn*	*39-434-32020*	*Pordenone Italy*	*1200*
2	*Fido Padova*	*39-49-663452*	*Padova Italy*	*1200*
3	*Fido PD 2*	*39-49-620035*	*Padova Italy*	*1200*
5	*The Wall*	*39-444-961708*	*Vicenza Italy*	*1200*
6	*Bit One*	*39-45-7730420*	*Verona Italy*	*1200*
7	*The Clivius Link*	*39-45-565988*	*Verona Italy*	*1200*
8	*Opus-Plat-One*	*39-431-438-271*	*Bibione Italy*	*2400*
9	*Uni Opus*	*39-45-7731283*	*Verona Italy*	*1200*
334	*Nort-West Italy net*	*39-11-5765565*	*Italy*	*1200*
1	*Fido To*	*39-11-5765-565*	*Torino Italy*	*1200*
2	*Opus Montecastello*	*39-131-355506*	*Alessandria Italy*	*2400*
3	*Opus Tecnocity*	*39-11-4115173*	*Torino Italy*	*2400*
4	*Opus Eporedia*	*39-125-611624*	*IVREA Italy*	*1200*
5	*OpusLandia*	*39-11-290312*	*Torino Italy*	*1200*
6	*Torino net*	*39-11-538601*	*Torino Italy*	*1200*
7	*Primula Rossa*	*39-131-42467*	*Alessandria Italy*	*1200*
335	*Italia Centro Sud*	*39-971-35447*	*Italy*	*1200*
1	*Fido Pz*	*39-971-35447*	*Potenza Italy*	*1200*
2	*S. Maria CV Bbs*	*39-823-812533*	*S. Maria Italy*	*1200*
3	*Fido Roma*	*39-6-9035120*	*Roma Italy*	*1200*
4	*Opus The World BBS*	*39-81-7433830*	*Napoli Italy*	*1200*
5	*Civitavecchia BBS*	*39-766-22077*	*Civitavecchia Italy*	*1200*
6	*Alex Fido*	*39-6-7482648*	*Roma Italy*	*1200*

CONFRONTO TRA SOFTWARE

Questa tabella contiene caratteristiche e prestazioni relative a quattro diffusi programmi di comunicazione per Amiga. Nell'effettuare il confronto sono stati inclusi solo i dati più significativi. Occorre tenere presente che per sfruttare molte delle caratteristiche indicate (chiamata automatica, auto redial etc.) è necessario l'uso di un modem Hayes-compatibile. Di ogni programma è stata presa in esame la versione più recente disponibile. Tutti i programmi richiedono Kickstart 1.2 e supportano il multitasking (possono cioè, nei limiti della memoria disponibile, essere utilizzati in contemporanea con altre applicazioni).

sulla quale ci si può spostare con i tasti cursore per effettuare correzioni. Il tipo di terminale più diffuso è il VT 100.

- Grafica ANSI: molti Bulletin Board Systems funzionanti su computer Ms/Dos compatibili prevedono l'opzione che consente di sfruttare il set di caratteri grafico ANSI. Se il vostro programma di comunicazione lo prevede potete, attivando questa funzione, vedere visualizzate sul vostro schermo immagini grafiche a colori.
- Protocolli di trasmissione: per permettere il trasferimento di programmi eliminando la possibilità di errori generati dalla qualità della connessione è necessario utilizzare uno dei numerosi protocolli disponibili nella maggior parte dei programmi di comunicazione. Essi inviano il file da trasmettere suddividendolo in blocchi di 128 bytes (o multipli di 128 bytes) ed aggiungendo par-

SOFTWARE	Online! 2.1	Diga! 1.0	Amic .58	A-Talk II
Prodotto da	Micro Systems Software Inc.	Aegis Development	Pubblico Dominio	Oxxi Inc.
Prezzo	$ 69.95	$ 79.95	contributo volontario	$ 99
Protocolli di trasferimento	Xmodem, Ymodem Zmodem, Kermit Cserve, YBatch	Xmodem, Ymodem Kermit, Cserve Doubletalk	Xmodem, WXmodem	Xmodem, Ymodem Zmodem, Kermit YBatch
Controllo CRC	si'	si'	si'	si'
Log su disco	si'	si'	si'	si'
Stampa	si'	no	si'	si'
Review	si'	no	no	no
Tasti funzione	20	50	20	30
Emulazioni di terminale	VT100, VT102, VT52, TEK.	VT100, VT52, TEK.	VT52, VT100.	VT100, VT52, TEK, H19.
Grafica ANSI	si'	no	no	no
Colori	8	4	8 (ridefinibili)	8
Font opzionali	si'	no	si'	si'
Modo Chat	si'	no	si'	si'
Agenda telef.	40 numeri	illimitata	40 numeri	illimitata
Modo Host	no	si'	no	si'
Auto Chop files	si'	si'	si'	si'
Auto Redial	99 volte	1 volta	1 volta	infinito
Redial su piu' numeri diversi	si'	no	no	no
Script Files	si'	si'	no	si'
Interlacing	si'	si'	si'	si'
Dimensioni max. (col..x righe)	132 x 64	128 x 50	132 x 50	132 x 50

ticolari codici di controllo; nel caso un blocco risulti incompleto o fallisca questa verifica, esso viene automaticamente ritrasmesso. I protocolli di trasferimento più diffusi sono Xmodem, Ymodem, Zmodem e Kermit.

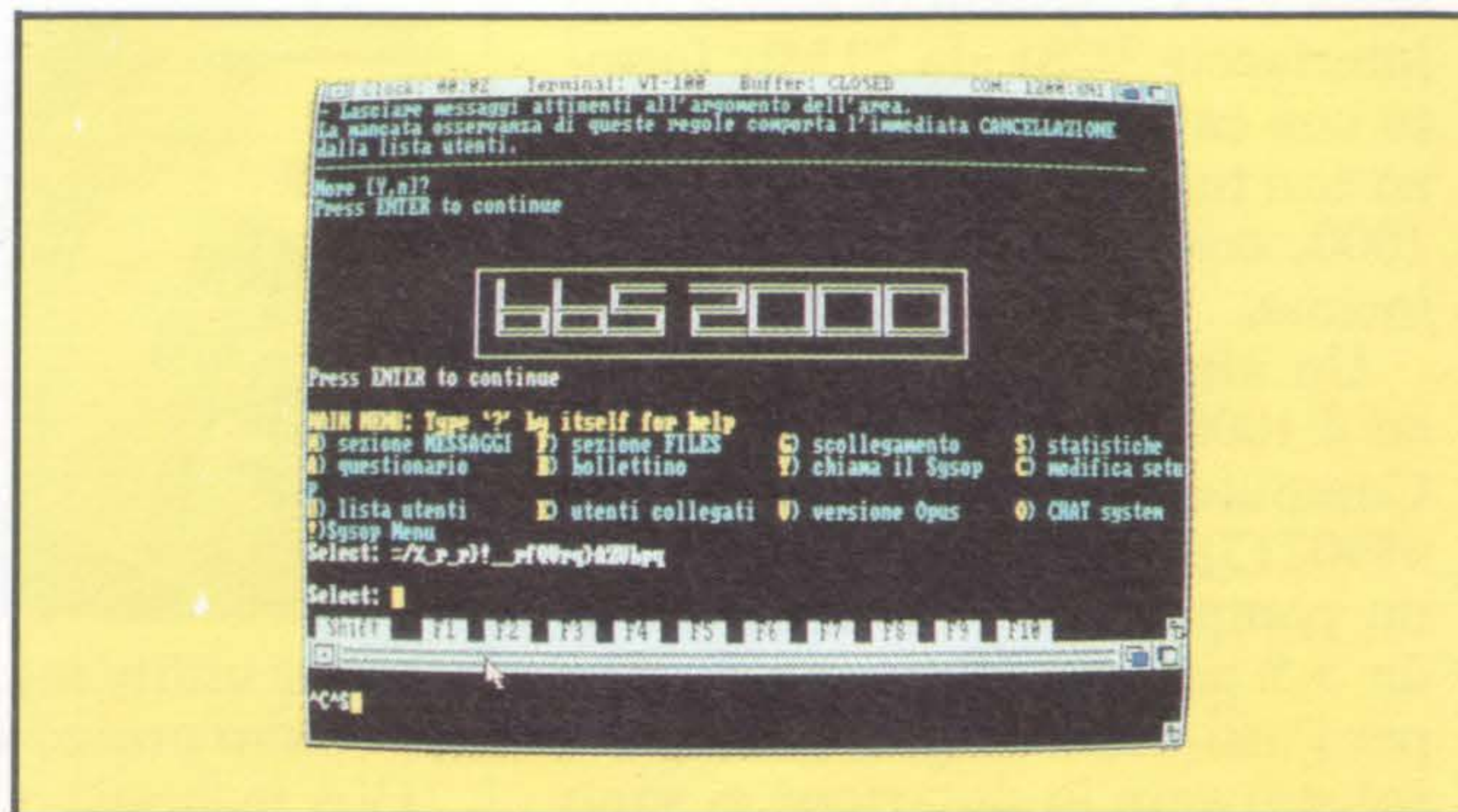
- Chat mode: denominata anche «split screen» (schermo diviso), questa funzione consente di dividere lo schermo in due finestre separate, contenenti una i caratteri da inviare e l'altra quelli ricevuti dal modem. È utile nel caso di collegamenti a sistemi che permettano il «chat» (conversazione simultanea tra utenti), in modo da rendere più chiaramente leggibile il testo.

I migliori programmi di comunicazione commerciali per Amiga sono il Diga (prodotto dall'Aegis) e l'Online! (della Micro Systems). Quest'ultimo in particolare, giunto alla release 2.01, è probabilmente uno tra i più completi software del suo genere mai apparsi sul mercato: oltre che presentare tutte le funzioni speciali precedentemente elencate, offre all'utente alcune opzioni irreperibili su altri programmi del genere per Amiga; ad esempio la pos-

sibilità di ridefinire la tavola dei caratteri Ascii a piacimento, permettendo di filtrare gran parte dei caratteri spurii generati dai disturbi sulla linea telefonica.

Anche il mondo del software di pubblico dominio mette a disposizione un nutrito catalogo di programmi di comunicazione, in genere di buona qualità; pur mancando della versatilità e dell'affidabilità dei programmi commerciali, quelli di pubblico dominio possono comunque essere utili per un primo approccio alle telecomunicazioni. Non mancano le eccezioni, come Access o Amic, due programmi di qualità addirittura superiore a quella di alcuni pacchetti software distribuiti commercialmente.

È bene ricordare che per collegarsi al servizio Videotel è necessario munirsi, oltre che di un modem V23, di un programma di comunicazione in grado di supportare il particolare set di caratteri grafico che esso utilizza, denominato VideoText. Tra quelli disponibili per Amiga ne ricordiamo due: l'inglese Ruby-View, prodotto dalla Y2 Computing, che lo commercializza unitamente ad una speciale interfaccia seriale da interporre tra il modem e la porta seriale di Amiga, e l'australiano Viatel, della Bruining Headlam Computers.

Su BBS 2000, tre aree nazionali ed internazionali Amiga-dedicate. Più di 200 programmi gratis!

News
DA TUTTO IL MONDO

COME AUMENTARE LA MEMORIA

ESPANSIONI ITALIANE

Finalmente disponibile in Italia una serie di espansioni di memoria di produzione nostrana a prezzi abbastanza accessibili. Vediamole.

Espansione da 0.5 a 1 Mb per Amiga 1000: non autoconfigurante (occorre un'apposita utility per farla riconoscere); si collega al bus esterno. Molto interessante anche il prezzo, 320 Klire per 512 Kb. Rivolgersi alla O.M.A. di Milano.

Espansione Newtronic da 2 Mb per Amiga 500: anche questa non è autoconfigurante e si collega allo slot laterale, ma 2 Mega sono davvero tanti. Costa 890.000 Lire.

Per l'Amiga 2000 è invece arrivata l'espansione ufficiale Commodore A 2052, 2 Mb di memoria da inserire in uno degli slot interni. Per 650.000 Lire avrete della sana memoria autoconfigurante e, una volta installata l'espansione, potrete dimenticarvene, cosa che invece non faranno i programmi «RAM-hungry» quali i CAD. Tutti i prezzi si intendono IVA inclusa.

HARD DISK

Mega, mega e ancora mega

PER QUALCHE MB IN PIÙ

Se avete necessità di immagazzinare velocemente grosse quantità di dati ecco due prodotti che fanno al caso vostro: l'Hard Disk della Xebec System di Roma, con interfaccia SCSI, da 20 Mb; fornito con cavi e software per l'utilizzo con tutti i modelli Amiga 500 e 1000, costa 1.250.000 Lire, IVA inclusa.

Un altro winchester per il 500 ed il 1000 è quello prodotto dalla Computer Center di Milano (02/4890213): offre 20 Mb di spazio in un compatto cabinet (meccanica da 3,5 pollici). Il disco è pronto per l'uso, essendo già formattato; sul software in dotazione ci sono molte utility interessanti.

Il suo prezzo è di 1.250.000 Lire IVA inclusa.

Play Game

In arrivo dagli States tre mega giochi marcati Micro Illusions: «*Land of Legends*», un role playing game giocando il quale sarete protagonisti di un mondo popolato da guerrieri, maghi e... preti. Gli appassionati dei motori invece troveranno eccezionale «*Turbo*», un velocissimo arcade di chiara ispirazione automobilistica. Notevole la possibilità di correre con un tuo amico via modem, sparando all'impazzata per liberare la strada da macchine agguerritissime. Infine, un programma dedicato a tutti gli aspiranti astronomi, «*Planetarium*»: tra le sue caratteristiche, degne di nota la presenza di oltre 9000 stelle fino alla settima magnitudine ed una completa visione della volta celeste, basata sugli ultimi dati forniti dalla NASA.

3M DATA DISPLAY

Lavagna luminosa

La 3M Italia (tel. 02/75451) ha posto in vendita uno schermo a cristalli liquidi che permette di proiettare, tramite lavagna luminosa, le immagini generate da un computer e, di conseguenza, anche le animazioni.

Va collegato direttamente all'uscita video (utilizzando il segnale Videocomposito o RGB TTL, entrambi compatibili con Amiga), ed è di uso molto semplice. Raccomandato per la didattica o le conferenze, il suo prezzo si aggira intorno ai 2.400.000 lire.

USA BEST

AMIGA DEV-CON

Ultimissime da Washington dove si programma il futuro.

Come di consueto, anche quest'anno ha avuto luogo a Washington la manifestazione AMIGA DEV-CON (Developers Convention), un congresso dedicato dalla Commodore statunitense ai programmatori e sviluppatori di software e hardware Amiga.

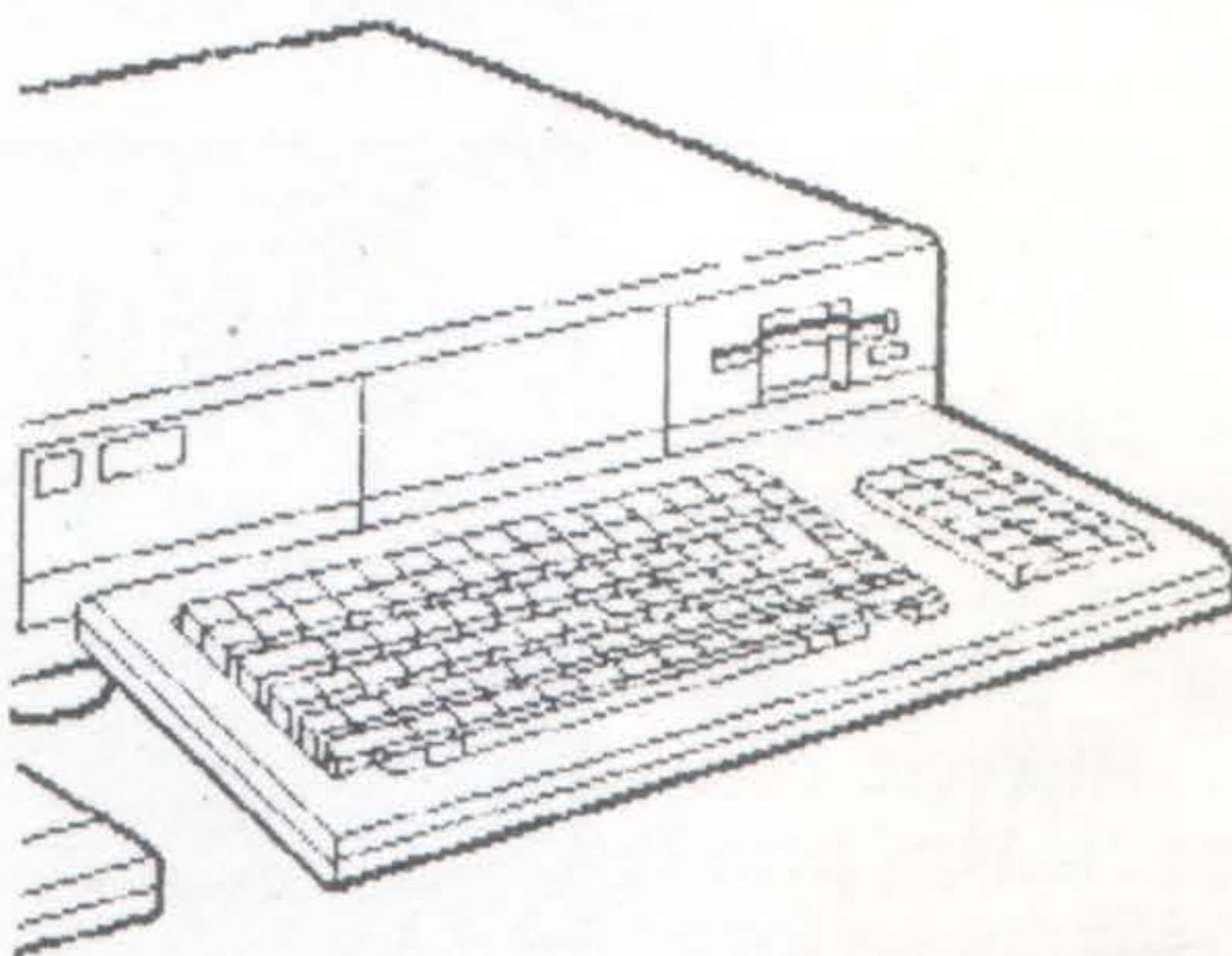
Hanno partecipato al congresso programmatori provenienti da ogni parte del mondo, dall'Australia alla Finlandia, dall'Italia alla Turchia (che, secondo alcuni dati diffusi nel corso della manifestazione, è il Paese in cui attualmente si registra il più rapido incremento nella domanda di prodotti Commodore).

Alcune tra le più note personalità dell'ambiente hanno tenuto conferenze e presentato le loro ultime creazioni: tra loro R.J. Mical, uno dei componenti del team che ha sviluppato il software del Workbench, e John Toebes della Software Distillery, la software-house nota per aver creato il Blink, il diffuso linker adottato anche dalla Lattice nelle versioni più recenti del suo compilatore C.

Causa questo carattere elitario della riunione, non orientata alla promozione di novità presso il pubblico degli utenti finali, non sono state presentate nuove macchine o accessori; in compenso sono stati annunciati due nuovi prodotti, denominati A-2500.AT e A-2500.UX. Non si tratta comunque di nuovi modelli di computer, ma di pacchetti hardware e software comprendenti le nuove schede Janus AT e Unix, da tempo annunciate, ma non ancora disponibili sul mercato.

Grande rilevanza ha assunto la presentazione delle caratteristiche della nuova versione del sistema operativo, il Kickstart 1.3. Non è ancora terminata la versione definitiva, agli sviluppatori è stata distribuita una «Gamma release» ma già si parla di una prossima release 1.4 ed addirittura di una futura 1.5!

Per il momento questi progetti sono ancora in fase embrionale, ma alcune caratteristiche sono già delineate: il Fast System, già implementato nella versione 1.3 per rendere più veloce la gestione di un hard-disk, dovrebbe essere adottato per tutti i device; la gestione dei comandi AmigaDos dovrebbe essere velocizzata; si parla di implementare una nuova versione della translator. Library; dovrebbero essere inoltre potenziate le capacità grafiche, grazie anche all'adozione di nuove versioni dei chip Agnus, Denise e Gary, permettendo una risoluzione di 640 x 400 pixel senza ricorrere all'interlace, con 4 colori visualizzabili contemporaneamente e selezionabili da una palette di 64 colori. Il condizionale è d'obbligo, poiché il progetto subirà quasi certamente qualche modifica in fase di realizzazione, con l'aggiunta di altre caratteristiche.

È stata analizzata in dettaglio la struttura dei files IFF, e sono state definite alcune specifiche di questo formato che ormai si è consolidato come standard, adottato dalla quasi totalità dei programmatori. Presentato inoltre un nuovo formato IFF compatibile denominato PGTB (ProGram Trace Back), destinato agli sviluppatori di software. Questo formato, unitamente ad un software di gestione apposito, permette la creazione di un file su disco contenente lo stato del sistema ed altre informazioni durante

l'esecuzione di un programma, in modo da individuare le cause del problema nell'eventualità di una Guru Meditation o di metterne alla prova il corretto funzionamento in varie situazioni. Un accorgimento di scarsa rilevanza per l'utente finale, ma molto utile per il debugging del software in fase di sviluppo.

Infine, una notizia che dovrebbe tranquillizzare i possessori di Amiga preoccupati per il futuro dei loro computer: l'introduzione di nuove macchine non comporterà l'uscita di scena dei modelli attuali. I progettisti di Amiga hanno assicurato che, nell'eventualità di modifiche hardware o dell'uscita di modelli più avanzati, sarà mantenuta la compatibilità totale con le macchine pre-esistenti.

MUSICA

Due ghiotte novità

Interessanti novità per i musicisti professionisti: l'FB-01 Datafiler, che permette un editing completo delle risorse dello Yamaha FB-01, comprese quelle non reperibili sul pannello frontale; è possibile editare, organizzare ed archiviare tanto le voci quanto le configurazioni.

Il programma è molto amighevole, ed aggiorna inviluppi ed algoritmi direttamente sullo schermo; può lavorare inoltre in multitask con un sequencer, e monitorare i cambiamenti eseguiti in tempo reale. È reperibile al prezzo di 90 dollari presso Triangle Audio, P.O. Box 1108, Sterling, VA 22170.

Lo Yamaha TX81Z editor/librarian offre invece la possibilità di modificare tutti i parametri del TX, ed anch'esso permette un «voice editing» interattivo; costa 79 dollari e si può ordinare a MidiTalk, P.O. Box 69, Albany, NY 12260.

(segue da pag. 37)

lare», atterrando quindi a bassa velocità ed evitando così di riprendere quota dopo un primo rimbalzo sulla pista (accade più facilmente di quanto pensiate). Questa manovra si chiama «flare».

La vostra velocità, dopo la richiamata, comincerà a diminuire, mentre diminuirà anche il vostro rateo di discesa (prima era di ben 1700 piedi al minuto); la velocità di atterraggio sarà intorno ai 50 nodi. Assicuratevi di essere allineati, in quanto il timone di direzione deve essere ormai diritto: esso è collegato con il ruotino, ed un atterraggio a ruotino sterzato porta ad immaginabili conseguenze. Quando toccherete terra sentirete il rumore caratteristico; l'indicatore di posizione dell'equilibratore (cioè del timone di quota) sarà quasi tutto su, «a salire»; portatelo subito «a scendere» (anche questo vi impedirà di rimbalzare se per caso sarete atterrati un po' troppo veloci) e premete ripetutamente l'1 per frenare. Siete arrivati.

È consigliabile, prima di cominciare la discesa, salvare la posizione su ram; in questo modo, se la situazione dovesse... precipitare, non sarete costretti a ripetere daccapo tutto il volo (che sarà durato circa 15 minuti), e comunque potrete allenarvi ad atterrare ripetutamente, magari variando (ma di poco) alcuni parametri per osservarne gli effetti. Il passo ulteriore è quello rappresentato dall'uso dei radioaiuti per la navigazione aerea; questo è un campo molto vasto, in cui dovreste esercitarvi con i dettagliati esempi del manuale. Ecco comunque alcuni cenni su come servirsi dei radioaiuti per la prima volta.

I Radiofari che utilizzeremo sono i VOR (Vhf Omnidirectional Radio Range — radiosentiero omnidirezionale), e sono quei cerchi graduati che si vedono sulle carte nautiche, con indicati nome, frequenza ed eventuale presenza del DME (Distance Measuring Equipment). Trasmettono segnali radio che indicano la loro posizione, su una rosa di 360 gradi (i cosiddetti «radiali») e con segnali diversi per ogni grado. Il pilota è quindi in grado di conoscere la propria posizione relativa, leggendola sugli strumenti VOR1 e/o VOR2. Ora un po' di pratica, nel modo più semplice: scegliamo sulla carta un aereoporto fornito di VOR-DME, in questo caso San José, a sud est di Oakland. Impostiamo sulla radio NAV1 la frequenza del radiofaro (che può essere captato fino ad una distanza di 30 - 100 miglia), che è di 114.10, clickando direttamente sulla radio. Quando la frequenza sarà impostata, vedremo il VOR1 animarsi. Clickate ora sul tasto contrassegnato dalla lettera V per cambiare la selezione di rotta fino a che non appare la scritta TO (verso) e fino a quando l'ago verticale non è perfettamente centrato. La rotta da seguire risulterà di 134 gradi e sul DME leggeremo la distanza di 28.8 miglia, quella che ci separa dalla nostra meta.

Dopo il decollo prenderemo quindi una prua di 134 gradi e durante il volo clickeremo sulla V per aggiustare gli spostamenti dell'ago e modificheremo di conseguenza la nostra rotta. Adesso però che avete rotto il ghiaccio con questo complicato programma, avrete anche forse un atteggiamento diverso nei confronti del manuale e potrete consultarlo con profitto anche nelle sue parti più difficili. ■

Dispense e programmi per Amiga, Cambio. Sono particolarmente interessato alla grafica - titolatori e Fish disk. Inviatemi le vostre liste, farò altrettanto. Il mio indirizzo è: Simone De Negri, via Puggia 22, 16131 Genova (GE). Massima serietà. Annuncio sempre valido.

Ho a disposizione programmi per Amiga, più di 600 programmi. Per ulteriori informazioni telefonare a: Alessandro Maggi, via Manin 12, 20013 Magenta (MI), tel. 02/9799448.

Dispongo di moltissimi programmi per Amiga, sempre in arrivo. Sono interessato inoltre a tutte le utility matematiche reperibili sul mercato. Scrivi o telefona nelle ore serali a:
Adolfo Salomone, via Guerrini 2, tel. 02/238147.

Amiga 500 con mouse e manuali originali compro a L. 600.000. Scrivere o telefonare a: Roberto Bettati, via Meucci 17, 20010 Marcallo (MI), tel. 02/9760125.

Club per utenti CBM64/128 e Amiga si è formato, per scambio di programmi senza scopo di lucro. Per ulteriori informazioni scrivere o telefonare a: Commodore club compagnia, via De Filippis 61, 84010 Cava Dei Tirreni (SA), tel. 089/464288.

Meteosat con apparecchi radio, cerco informazioni su possibilità di ricezione, ed altre applicazioni analoghe. Chi sapesse qualcosa scriva o telefoni a: Luca Signorelli, via Fabio Filzi 15, 37045 Legnago (VR), tel. 0422/21819.

Attenzione!! Scambio programmi e manuali, tratto solo programmi selezionati. Scrivere o telefonare a: Fabrizio Italia, via Palestro 101, 96014 Floridia (SR), tel. 0931/941645.

Cambio software, massima serietà, astenersi speculatori. Annuncio sempre valido. Scrivere: Stefano Manca, via Parigi 6, 07100 Sassari (SS), tel. 079/218987.

Programmi novità per Amiga, Arrivi settimanali da Germania e Stati Uniti. Supporti magnetici, hardware, new-niki 2, penna ottica, speeddos-plus-new. Per informazioni telefonare o scrivere a: Giuseppe Borracci, via Mameli 15, 33100 Udine (UD), tel. 0432/580157.

Le ultime due pagine contenevano gli occhiali 3D da ritagliare

Le ultime due pagine contenevano gli occhiali 3D da ritagliare